DOCUMENT RESUME

ED 091 929 FL 005 419

AUTHOR	Josselson, Harry H.
TITLE	Development of a Russian Idiom List. Final Report.
INSTITUTION	Wayne State Univ., Detroit, Mich. Dept. of Slavic Languages.
SPONS AGENCY	Institute of International Studies (DHEW/OE), Washington, D.C.
BUREAU NO	9-7710
PUB DATE	Jul 72
CONTRACT	OEC-0-9-097710-3733(014)
NOTE	85p.; Idiom list is available from Wayne State University Press
EDRS PRICE	MF-$0.75 HC-$4.20 PLUS POSTAGE
DESCRIPTORS	Computational Linguistics; Computer Programs; *Idioms; Information Processing; Language Research; Programing; *Russian
IDENTIFIERS	*National Defense Education Act Title VI; NDEA Title VI

ABSTRACT

This research is designed to fulfill the need of scholars, teachers, and developers of teaching materials who are active in the study and compilation of Russian textbooks. Through computer processing, the idioms were classified on the basis of pertinent linguistic criteria, including phonological, morphological, and syntactic information. Included in the present report is a description of the linguistic and computer operations used to extract the data and the three main programs developed during the course of the work. These programs are written in OS/360 assembler language. It is hoped that the results of this research will provide a foundation for later expansion of the idiom list and also a procedure by which both expansion of this list and compilation of other lists can be made possible through appropriate computer utilization. The actual idiom list is not included in this report. (Author/PM)

ED 091929

TITLE VI, NDEA, SEC. 602
Contract No. OEC-0-9-097710-3733
Bureau No. 9-7710

Final Report

Project No. 097710
Grant No. OEC-0-9-097710-3733 (014)

Dr. Harry H. Josselson
Wayne State University
Detroit, Michigan 48202

U.S. DEPARTMENT OF HEALTH,
EDUCATION & WELFARE
NATIONAL INSTITUTE OF
EDUCATION
THIS DOCUMENT HAS BEEN REPRO
DUCED EXACTLY AS RECEIVED FROM
THE PERSON OR ORGANIZATION ORIGIN
ATING IT. POINTS OF VIEW OR OPINIONS
STATED DO NOT NECESSARILY REPRE
SENT OFFICIAL NATIONAL INSTITUTE OF
EDUCATION POSITION OR POLICY

DEVELOPMENT OF A RUSSIAN IDIOM LIST

July 1972

EL005 919

U.S. DEPARTMENT OF HEALTH, EDUCATION, AND WELFARE

U.S. Office of Education
Institute of International Studies

ABSTRACT

TITLE OF PROJECT: Development of a Russian idiom list.

STATEMENT OF PURPOSES: The purpose of this research has been to compile and distribute a Russian idiom list, which presently does not exist, to fulfill the needs of scholars, teachers, and compilers of teaching materials who are active in the study and compilation of Russian textbooks.

PROCEDURES: The developed procedures utilized the results of four years research, sponsored by the National Science Foundation, which has produced a collection of coded linguistic information compiled from two authoritative Russian language lexicons. This information is presently stored on magnetic tape at Wayne State University and with all of the lexical entries in both lexicons needed for idiom analysis appropriately marked on tape, a comprehensive list of idioms was extracted by computer processing. This processing provided classification of the idioms on the basis of pertinent linguistic criteria, including phonological, morphological and syntactic information. This processing produced a list of idioms alphabetized and cross-referenced for printing, which is being published by Wayne State University Press.

CONTRIBUTION TO EDUCATION: The results of this research will be a significant contribution to all educators, linguists, scholars, and compilers of teaching materials of Russian who profess a need and a desire for a list of Russian idioms for use in the study and compilation of Russian textbooks. Further, the results will provide a foundation for later expansion of the list and also a procedure by which both expansion of this list and compilation of other lists can be possible through appropriate computer utilization.

Final Report

Project No. 097710
Grant No. OEC-0-9-097710-3733 (014)

DEVELOPMENT OF A RUSSIAN IDIOM LIST

Dr. Harry H. Josselson

Wayne State University

Detroit, Michigan 48202

July 1972

The research reported herein was performed pursuant to a grant with the Office of Education, U.S. Department of Health, Education, and Welfare. Contractors undertaking such projects under Government sponsorship are encouraged to express freely their professional judgment in the conduct of the project. Points of view or opinions stated do not, therefore, necessarily represent official Office of Education position or policy.

FL 005 419

U.S. DEPARTMENT OF HEALTH
EDUCATION & WELFARE
NATIONAL INSTITUTE OF
EDUCATION

U.S. DEPARTMENT OF
HEALTH, EDUCATION, AND WELFARE

U.S. Office of Education
Institute of International Studies

TABLE OF CONTENTS

Page

Introduction 1

Procedure. 3

Computer Processing. 7

Conclusion 9

Appendix

Table A: Stylistic Evaluation. 11

Table B: Field of Usage. 13

Table C: Explanation of Symbols and Abbreviations Used in Printouts. 16

Table D: "Idiom Updater" Program 18

Table E: "Cross-Reference" Program 50

Table F: "Idiom Formatter - Main Routine". 70

Introduction

1. The goal of the present study has been the compilation and distribution of a Russian idiom list, which presently does not exist, to fulfill the needs of scholars, teachers, and compilers of teaching materials who are active in the study and compilation of Russian textbooks.

This study is an outgrowth of a research project, sponsored for four years by the National Science Foundation, which has produced a collection of coded linguistic information compiled from two authoritative Russian language lexicons, i.e. The Толковый словарь русского языка, edited by D. N. Ušakov, published in Moscow in 1935; and the Словарь русского языка, published in 1957 by the Academy of Sciences of the U.S.S.R. in Moscow. Among the coded linguistic information now on tape for each entry in the two source lexicons mentioned above, is an indication of its participation in an idiom. Each lexical entry which had been marked for participation in an idiom was subjected to a processing which will be described below. Due to the dissimilarities existing among word sequences marked as idioms by the source lexicons, a classification system was established to distinguish between true idioms and complex lexical entries, e.g. железная дорога - railroad -. Thus the present list contains word sequences of both types. However, they were submitted to a different type of processing and are clearly distinguished in the list.

1.1 The linguistic and computer operations which resulted in the present list of Russian idioms can be summarized in the following steps:

1.1.1 Compilation of a list of word sequences marked in the source lexicons as idioms of the Russian language.

1.1.2 Investigation and specification of the limitations or restrictions on usage of these idioms. These restrictions may be external, i.e. situational - inherent in the semantic and stylistic, as well as cultural considerations, or internal, i.e. inherent in the grammatical properties of the constituents of which the idiom is comprised. Such specification may be represented in the standard record, the coding symbols used in recording the idioms, the Russian language examples, and the English equivalents.

1.1.3 Providing Russian language examples in order to illustrate the type of context in which a given idiom may appear, both in terms of its semantic content and its syntactic role in a sentence.

1.1.4 Classifying those word sequences extracted from the source lexicons, and marked by them as idioms, as to their lexicological status, i.e. distinguishing between true idioms and complex lexical units.

1.1.5 Establishing the English equivalents for those word sequences classified under step four as true idioms.

1.2 Steps one and two were applied to both source lexicons. However, upon examination of the data processed in these two steps, it was determined that there was no substantive difference between the data extracted from the two lexicons. Therefore, only the data from the

Словарь русского языка, published by the Academy of Sciences of the U.S.S.R., was subjected to steps three through five. The output of the finished work consists of two printouts: one of the data extracted from the Толковый словарь русского языка, processed through step two; and one of the data extracted from the Словарь русского языка, processed through step five.

1.3 The two lexicons cited above were the only source of the Russian idioms together with an indication of their grammatical and stylistic properties. However, the information provided in steps three and five, i.e. the Russian language examples and the English equivalents, was drawn from several sources. The language examples were taken from the source lexicon whenever they were provided. Where the source lexicon did not show a language example for a given idiom, a member of the research staff with a native competency in Russian, Mrs. Nina Anderson, provided a language example of her own composition.

Appropriate English idiomatic equivalents or explanations for each Russian idiom were selected by a more eclectic and less systematic procedure. The sources of English equivalents were the following:

1.3.1 Members of the research staff, including: Dr. Alexander Vitek, Joan McLetchie, Ronald Stauffer, Immo Knoff, Amelia Steiger.

1.3.2 Highly educated native speakers of Russian with a high degree of competency in literary English: Dr. Vera S. Dunham, Dr. Assya Humesky.

1.3.3 Various lexicons:

Dictionary of Spoken Russian, New York: Dover Publications, Inc., 1958.

Henderson, B.L.K., A Dictionary of English Idioms, London: James Blackwood & Co, LTD., 1964.

Jaszczun, Wasul and Krynski, Szymon, A Dictionary of Russian Idioms and Colloquialisms, Pittsburgh: University of Pittsburgh Press, 1967.

Sergievsky, Nicholas N., Idiomatic Russian, New York: International Universities Press, Inc., 1967.

Smith, L.P., Words and Idioms, Boston and New York: Houghton Mifflin Company, 1925.

Stein, Jess (ed.), The Random House Dictionary of the English Language, New York: Random House, Inc., 1969.

Wood, F.T., English Colloquial Idioms, London: MacMillan and Co.LTD., 1970.

Procedure

2. The compilation of the list of Russian idioms described above and the analysis of the idioms was conducted in the following three stages:

2.1 In the first stage of research a list of word sequences, including both idioms and complex lexical entries, was extracted from the source lexicon. During this process, each word sequence was subjected to the following coding procedures:

2.1.1 Each word sequence was recorded on a specially designed coding sheet and was assigned an I.D. number and a head word, which corresponded to the canonical entry under which the word sequence appeared in the source lexicon.

2.1.2 In addition to a numeric I.D. number, a standard record was established for each word sequence which consisted of a series of numeric codes representing the following information:

2.1.2.1 The number of the word sequence in relation to the canonical entry under which it appeared in the source lexicon, i.e. whether it was the first, second, third, etc. of such word sequences listed under the given canonical entry.

2.1.2.2 Whether or not an explanation for the given word sequence is shown under the given canonical entry.

2.1.2.3 Whether or not a language example for the given word sequence is shown under the canonical entry.

2.1.2.4 The stylistic status[1] of the word sequence, e.g. colloquial, archaic, etc., wherever such information is supplied by the source lexicon. (see Table I)

2.1.2.5 The field of usage[2] of the word sequence, e.g. mathematical, medical, wherever such information is supplied by the source lexicon. (see Table II)

2.1.2.6 How the given word sequence was set off by the source lexicon, i.e. whether it was preceded by a diamond ◊, the convention used by the source lexicon to indicate an idiom, or by the words "В выражении" (in the expression...) or "в сочетании" (in the combination...)

2.1.3 In this first stage of research, each word sequence, in conjunction with the head word under which it appears, was classified as belonging to one of two types. In most instances a given word sequence is entered under every canonical entry which serves as a key constituent of the word sequence. Thus, the word sequence Держать камень за пазухой (to hold a grudge) is entered under all three key constituents: держать, камень, пазуха. However, an explanation is provided for this idiom only under

---

[1] Referred to as "status labels" by Einar Haugen, Norsk Engelsk Ordbook, Oslo: Universitetsforlaget, 1965.

[2] Referred to as "subject labels" by Einar Haugen, Ibid.

the canonical entry камень. Under the entries держать and пазуха, the reader is referred to камень. Therefore in the process of compilation of the list of idioms, any given word sequence consisting of two or more key constituents usually appeared more than once. In the coding process, each appearance of a given word sequence, in conjunction with the canonical entry under which it appeared, was classified as belonging to one of two types:

2.1.3.1 A source entry - under this canonical entry, the given sequence appeared together with an explanation,

2.1.3.2 A cross-reference - under this canonical entry, the given sequence appeared together with a reference to another canonical entry under which an explanation of the word sequence is provided.

In the source lexicon, the system of cross-referencing was not consistently applied to every word sequence designated by the source lexicon as an idiom, expression, or word combination, i.e. a given word sequence was not necessarily entered under each of its key constituents. Therefore, a coding system was devised which resulted in the automatic generation of cross-references for each key constituent of a word sequence, excluding the key constituent under which the explanation was given. Thus, for the idiom держать камень за пазухой, a cross-reference of this idiom was generated under the head word держать and under the head word пазуха, but not under камень, which is the source entry for this idiom.

2.1.4 In recording the word sequences themselves on the coding sheets, a set of symbols was used as a coding system in which to express a grammatical and lexical analysis of each word sequence.

2.1.4.1 As regards the lexical content of the idioms, there are two basic types:

1) absolutely constant idioms
2) partially constant idioms

2.1.4.1.1 The first type of idioms may be illustrated by the idiom держать камень за пазухой (to hold a grudge) which does not permit any lexical variants ("lexical variants" implies replacement of lexical constituents).

2.1.4.1.2 The second type is represented by such idioms as держать камень шире (don't hold your breath', literally 'to hold the pocket wider') in which the last constituent, i.e. шире ('wider') permits the variant по-шире ('a little wider') without any change in the meaning of the idiom. Another example of the latter type is the idiom прижать к стене~стенке ('to back someone against the wall') in which the variable constituents are freely interchangeable without any influence on the semantic value of the idiom. This type of lexical variation is symbolized in the following way: прижать к $стене/стенке$.

The idiom прижать к $стене/стенке$ may be used to exemplify another type of lexical variation. The verb прижать is transitive and requires a complement in the accusative case. The semantics of the idiom прижать к $стене/стенке$ restricts the lexical items capable of functioning as

complements of прижать to substantives containing the meaning "person". This restriction is symbolized by placing the interrogative кто (i.e. who) in brackets in the appropriate case form, i.e. [кого]. In instances where any lexical item belonging to a specific word class can fill a given slot in an idiom, this is symbolized by placing the appropriate interrogative pronoun (or pronouns) in brackets. Thus, words in brackets represent "replacive" words, i.e. in the actual usage of a given idiom, a word which appears in the idiom list in brackets will be replaced by another word belonging to the same word class and in the same inflectional form. For example, in the idiom прижать [кого] к $стене/стенке$, [кого] might be replaced by студента and the actual idiom might be: учитель прижал студента к стене. (The teacher backed the student against the wall.)

A third type of lexical variation is exemplified in the idiom по (мере) возможности (i.e., 'as far as possible'; 'as much as one can'; literally, 'according to the degree of possibility'). Wherever a lexical constituent of an idiom is optional, i.e. it can be omitted without altering the meaning of the idiom, that lexical constituent is placed in parentheses. Thus, in the given idiom, the lexical constituent мере (degree) is optional.

Another type of partially constant idiom is one containing a grammatical variation, i.e. the inflectional form of a specific lexical constituent of a given idiom may vary without altering the semantic content of the idiom. As with lexical variation, this is expressed by enclosing the variables in dollar signs. However, the relation between the two forms, that of grammatical variability, is indicated by the sign <, e.g.: вилами $на воде<по воде$ $писано/написано$ (i.e. 'it's still up in the air'; literally, 'It is written/written down in the water/along the water by a pitchfork'). As is exemplified here, more than one type of variation can occur within a given idiom.

2.2 The second stage of research involved proof-reading and up-dating of the idiom list which had been compiled in the first stage. Preliminary to this, two printouts were made, one of all the source entries ordered alphabetically and one of all the cross-references ordered alphabetically. Each printout was proof-read for spelling and stress errors.

2.2.1 In the printout of cross-references, wherever a generated cross-reference represented a duplicate of an original cross-reference, it was removed. In those instances in which only a generated cross-reference existed for a specific lexical constituent of a given idiom, due to an oversight on the part of the source lexicon, it was retained.

2.2.2 In the printout of source entries, three additions were made to each idiom. Wherever the standard record of a source idiom indicated that an example was shown for that specific idiom in the source lexicon, the example was printed in the appropriate place on the printout and stress marks were added to each polysyllabic word in the example. Where no example was given in the source lexicon, a native speaker of Russian, Mrs. Nina Anderson, added an example of her own composition. Secondly, each word sequence was classified as a true idiom or as a complex lexical unit; expressions such as железная дорога - 'railroad' - and приусадебный участок - 'privately owned collective farm land' - were classified as complex lexical units. This classification of the word sequences into

two types was symbolized by the addition of the arabic numeral seven to the standard record of complex lexical units. The third addition to the idiom record in stage II was made only to those word sequences not marked seven, i.e. to the true idioms. Wherever possible each Russian idiom was supplied with one or more English equivalents. These English equivalents are of two basic types:

a) English idioms
b) English explanations

In most instances, one or more English idioms were found which corresponded in meaning to a given Russian idiom. Sometimes the English equivalent not only corresponded in meaning to the Russian, but corresponded to it almost word for word, e.g. по сердцу [кому] - after one's own heart -. In other instances there was no literal correspondance between the Russian and the English, e.g. Ни в сказке сказать, ни пером описать; literally - you can't tell it in a story, you can't describe it with a pen -. The English equivalents for this idiom are - out of this world; like nothing you've ever seen; words can't describe it; beyond description -.

Where no corresponding English idiom was found, an English explanation of the Russian idiom was supplied in brackets, e.g. (Всегда) виноват стрелочник; literally - the switchman is always guilty -. [Refers to situations in which an underling is blamed and becomes the scapegoat.]

2.3 The third stage of research entailed combining the source idioms and cross-references. A new printout was produced in which the idioms, both source idioms and cross-references were listed alphabetically, together with their standard records, I.D. numbers and, for source idioms, English equivalents and Russian language examples, under the appropriate head words which were also ordered alphabetically. Thus, under a head word, an indefinite number of idioms of one or both types may appear. For example, two idioms are listed under the head word суета; мышиная суета which is a cross-reference and суета сует which is a source idiom.

Both of these printouts, the source printout and the cross-reference printout, about 1000 pages each, are stored in the Slavic Department of Wayne State University, and are available either on magnetic tape or printout form.

Computer Processing

3. The idioms are gathered in a data file which is currently maintained on magnetic tape. Each entry contains the following fields:

the ID, unique to each entry (see sec. 2.1.1)
the head word (see sec. 2.1.1)
the standard record (see sec. 2.1.2)

All of the preceding fields are of fixed length; all of the following are of variable length, and are preceded by a count of the number of characters in the field:

the Russian idiom,
the English equivalent,
the language examples,
an extra field, for possible future use.

The extra field is at present always empty, that is, with a count of zero. In any given record, the equivalent or examples may also be empty.

Three main programs were developed during the course of the work: the idiom updater, the cross-reference program, and the format program. These programs were written in OS/360 assembler language; the source listings follow in tables D, E, and F.

3.1 The idiom updater makes changes to the idiom file using a file of card images on which the changes to be made have been coded, called the update file. Both the idiom file and the update file are in ID order. The two files are merged; if the next idiom ID is less than the next update ID, the idiom is not changed, and the next idiom is obtained; if the idiom ID is greater than the update ID, the update is an insertion of a new entry, after which the next update is obtained; if the ID's match, the update is a change to the idiom record, after which both a new update and new idiom must be obtained.

3.1.1 Changes can be made to the idiom records at three levels. 1) Entire records can be inserted or deleted. 2) The fields of a record, as enumerated above, may be replaced as a unit. 3) Individual elements in the variable fields may be inserted, replaced, or deleted. The elements of the variable length fields are, roughly, strings of letters (that is, words), punctuation marks, or special characters.

3.2 The cross-reference program generates new idiom entries, and these, along with the original entries, form the output file. One new entry is generated for each special cross-reference character found in the idiom field of the record. Such characters are associated with a word in the idiom, and this word becomes the head word of the new entry. There are four cross-reference characters.

3.2.1 An underscore (_) precedes the word which is to be a new head word, and implies that the word is in canonical form; e.g. the infinitive for verbs, masculine nominative singular for nouns and adjectives, etc.

3.2.2 An asterisk (*), which also precedes the cross-referenced word, implies that the word is not in canonical form. As a result, the new head word is flagged with the asterisk. Part of the development of the idiom file was to replace this cross-reference combination with a canonical form.

3.2.3 Such replacement involved the use of the third cross-reference character, a colon (:), which follows the word in non-canonical form, and is in turn followed by the new head word in canonical form. The colon and following canonical form do not appear in the listing.

3.2.4 The fourth character, the logical-or sign, (|), precedes the reference word, the word following the English "R". The resulting new entry has as head word this reference word, and does not contain the reference. Thus, a source entry is generated from an original (or recorded) cross-reference. The purpose of this step was to check the consistency of the cross-referencing, and this character was not used in later stages of the work.

3.2.5 An idiom generated from a source idiom will contain, in addition to the original, the reference to the cross-referenced word. An exception to this rule applies to cross-referenced words within groups of variants. There the idiom in the generated entry will include the original idiom plus the reference, except the variant containing the cross-referenced word replaces the group of variants. If the original idiom is a cross-reference, the original reference is retained in the generated idiom.

3.2.6 To accomplish this procedure, the program has two phases. The first phase scans the idiom to determine for each cross-reference whether it occurs in a group of variants, and if it does, which variant (called a sub-group). The group and sub-group are identified by the number of the element with which they begin. The second phase uses this information to construct, element by element, the generated idiom for each cross-reference character.

3.2.7 In addition to the fields of the idiom file, the output file of the cross-reference program contains fields for sorting in other than ID order, for example, alphabetically by head word or idiom, source vs. cross-reference, etc.

3.3 The format program arranges the output from the cross-reference program in readable form. It is assumed the cross-reference output has been sorted using the head word, and perhaps other fields of the record, as a key. The head word is only printed once for each group of entries with the same head word. The standard record, idiom, English equivalent, and language examples each begin a new line.

3.3.1 In order to provide a multi-column format, a two-dimensional array is used, in which the elements are the lines of each column.

3.3.2 Three subroutines are used so the elements can be saved in the array column-order first, but printed row-order first. The first establishes how much of the longer fields will fit on one line of a column. The second returns the next element in the column. This involves spacing, if any, inserting the head words, and checking for the end of the column. When all columns are full, it calls the third subroutine, which prints the array row by row (that is, line by line).

A fourth subroutine adds capitalization for the first alphabetic character of the idiom.

Conclusion

4. The results of the linguistic and computer operations described above are the two printouts described above which represent a compilation and analysis of the word sequences marked as idioms in the source lexicons, and are available either on magnetic tape or printout form from the Slavic Department at Wayne State University.

4.1 The publication and dissemination of a Russian idiom list based on this study will supply long needed material on the instructional level as well as provide an impetus for further research into the various linguistic aspects which are a characteristic property of idioms, relating them to the structure of the language as a whole.

4.2 The utilization of methods of computer processing for this research has insured a greater precision and objectivity of the results, contributing also to the improvement of methods previously used as well as to the development of new ones. In addition, the permanency of the data encoded in the process of this research permits their subsequent utilization in any further study of Russian idioms.

APPENDIX

TABLE A

Stylistic Evaluation
(cf. 2.1.2.4)
(code derived from both lexicons)

ABBREVIATION used in source lexicon	CODE	EXPANDED TERM
None	00	———
арго	01	арго
бран.	02	бранное
вульг.	03	вульгарное
высок.	04	высокого стиля
газет.	05	газетное
груб. прост.	06	грубо-просторечное слово/выражение
детск.	07	детское
дореволюц.	08	дореволюционное
загр.	09	за границей
иностр.	10	иностранное
ирон.	11	ироническое
сатир.	12	сатирическое
канц.	13	канцелярское
книжн.	14	книжное
ласкат.	15	ласкательное
моск.	16	московское
нар.	17	народное
нар.-поэт.	18	народнопоэтическое
неодобрит.	19	неодобрительное
неправ.	20	неправильное
неприл.	21	неприличное
нов.	22	новое
обл.	23	областное
офиц.	24	официальное
перен.	25	переносное
поэт.	26	поэтическое
презрит.	27	презрительное
пренебр.	28	пренебрежительное
простореч.	29	просторечие
публиц.	30	публицистика
разг.	31	разговорное
ритор.	32	риторическое
уничижит.-ласк.	33	уничижительно-ласкательное
старин.	34	старинное
торж.	35	торжественное
увелич.	36	увеличительное
укор.	37	укоризненное
уменьш.	38	уменьшительное
уменьш.-ласк.	39	уменьшительно-ласкательное
уничижит.	40	уничижительное
усил.	41	усилительное

ABBREVIATION used in source lexicons	CODE	EXPANDED TERM
устар.	42	устарелое
устар.-поэт.	43	устарелое поэтическое слово/выражение
уступит.	44	уступительное
утвердит.	45	утвердительная (частица)
фам.	46	фамильярное
уменьш.-уничижит.	47	уменьшительно-уничижительное
церк.-книжн.	48	церковно-книжное
школьн.	49	школьное
шутл.	50	шутливое
эвф.	51	эвфемистическое
ласк.-фам	52	ласкательно-фамильярное
увелич.-уничижит.	53	увеличительно-уничижительное
уменьш.-пренебр.	54	уменьшительно-пренебрежительное
журнал.	55	журнальное
церк.-слав.	56	церковно-славянское
прост. ирон.	57	просторечие, ироническое
разг.-устар.	58	разговорное, устарелое
List type	99	**e.g. stylistic evaluation only in one meaning, e.g.** под парами
устар.-прост.	59	устарелое, просторечие
книж.-ирон.	60	книжное, ироническое
книж.-устар.	61	книжное, устарелое

TABLE B

Field of Usage
(cf. 2.1.2.5)
(code derived from both lexicons)

ABBREVIATION used in source lexicons	CODE	EXPANDED TERM
None	000	———
авиац.	001	авиация и воздухоплавание
агр.	002	агрономия
анат.	003	анатомия
антр.	004	антропология
апт.	005	аптекарское
археол.	006	археология
архит.	007	архитектура
астр.	008	астрономия
бакт.	009	бактериология
банк.	010	банковское
биол.	011	биология
биох.	012	биохимия
бот.	013	ботаника
бухг.	014	бухгалтерия
ветер.	015	ветеринария
воен.	016	военное
геогр.	017	география
геодез.	018	геодезия
геол.	019	геология
геофиз.	020	геофизика
гидрол.	021	гидрология
гидротех.	022	гидротехника
горн.	023	горное
грам.	024	грамматика
дипл.	025	дипломатия
ест.	026	естествознания
ж.-д.	027	железнодорожное
живоп.	028	живопись
зоол.	029	зоология
искус.	030	искусствоведение
ист.	031	история
кино.	032	кинематография
кулин.	033	кулинария
лес.	034	лесоводство
лингв.	035	лингвистика
лит.	036	литературоведение
лог.	037	логика
маляр.	038	малярное дело
мат.	039	математика
мед.	040	медицина
метал.	041	металлургия
метеор.	042	метеорология
мех.	043	механика

ABBREVIATION used in source lexicons	CODE	EXPANDED TERM
мин.	044	минералогия
мор.	045	морское дело
муз.	046	музыка
науч.	047	научное
опт.	048	оптика
охот.	049	охотничье
палеон.	050	палеонтология
педаг.	051	педагогика
полигр.	052	полиграфия
полит.	053	политика
плотн.	054	плотничье дело
порт.	055	портновское дело
проф.	056	профсоюзный
псих.	057	психология
пчел.	058	пчеловодство
радио.	059	радиотехника
религ.	060	религия
ремесл.	061	ремесленное
рыб.	062	рыболовство, рыбоводство
сад.	063	садоводство
сапож.	064	сапожное дело
слес.	065	слесарное дело
социол.	066	социология
спец.	067	специальный термин
спорт.	068	спортивный термин
стат.	069	статистика
стол.	070	столярное дело
с.-х.	071	сельское хозяйство
театр.	072	театральный термин
тех.	073	техника
тип.	074	типографское дело
торг.	075	торговое
фарм.	076	фармакология; фармация
физ.	077	физика
физиол.	078	физиология
филол.	079	филология
филос.	080	философия
фин.	081	финансы
фот.	082	фотография
хим.	083	химия
хр. э.	084	христианская эра
церк.	085	церковное слово/выражение
шахм.	086	шахматный термин
экон.	087	экономика
электр.	088	электротехника
этногр.	089	этнография
этнол.	090	этнология
юр.	091	юридический термин/право
текст.	092	текстильный термин

ABBREVIATION used in source lexicons	CODE	EXPANDED TERM
воровск.	093	воровское
no abbreviation	094	радиотелеграф
no abbreviation	095	карточное
no abbreviation	096	студенческое/вузовское
no abbreviation	097	тюремное
no abbreviation	098	бильярдное
no abbreviation	099	библиотечное
no abbreviation	100	беговое и скаковое
no abbreviation	101	кинотеатральное
no abbreviation	102	военно-морское
микробиол.	103	микробиология
актер.	104	актерское
коннозавод.	105	коннозаводческое
фабр.	106	фабричное
строит.	107	строительное дело
миф.	108	мифология
no abbreviation	109	бурсацкое
no abbreviation	110	шахтерское
no abbreviation	111	арго грузчик
no abbreviation	112	учительское-арго
no abbreviation	113	семинарское
no abbreviation	114	больничное
List type	999	e.g. field of usage in one meaning only

TABLE C

Explanation of Symbols and Abbreviations Used in Printouts

I. Russian Idioms: The following symbols and abbreviations are to be interpreted thus:

a) Parentheses ( ) signal optional constituents, e.g. под (са'мым) но'сом.

b) A slash / ; elements separated by a slash are lexical variants and are mutually exclusive, e.g. наставить/натянуть.

c) A left arrow < ; elements separated by a left arrow are grammatical variants and are mutually exclusive, e.g. но'са<но'су.

d) Dollar signs $...$ indicate the left and right boundaries of a sequence of lexical and/or grammatical variants, e.g. $наста'вить/натяну'ть$ нос из-под (са'мого) $но'са<но'су$.

e) Equal signs =...= signal an element or elements with which a given idiom usually occurs, but which are not part of the idiom, e.g. с ми'ром =иди'/поезжа'й= .

f) INF. = infinitive
V. = verb
ETC. = et cetera

g) Brackets [...]; the following elements are enclosed in brackets:
1. pronouns, including prepositional phrases, e.g. нами'м, у кого', чей
2. INF., V.
3. conjunctions plus...; e.g. что...; чтобы... These elements, when enclosed in brackets, are "replacive" words, i.e. they indicate that a given slot in an idiom may be filled by any lexical item of a given word class or any given syntagma or clause having a specific syntactic construction, e.g. $наста'вить/натяну'ть$ нос [кому']. Where two or more such elements occur in a given idiom, a left arrow is used to indicate that they are mutually exclusive, and a plus sign + is used to indicate they are used together,e.g. не ме'сто [чему'< INF]; кровь [кого' + на ком].

h) R signals a cross-reference, i.e. the reader is referred to the head word following R, e.g. нали'ться кро'вью R нали'ться.

II. Idioms and examples: The following symgols occur in the idioms and the examples and are to be interpreted thus:

a) An apostrophe ' indicates stress and immediately follows the stressed element in a word or prepositional phrase, e.g. наста'вить; на' душу.

b) A period . indicates ё and immediately follows the letter е, e.g. в че.м.

c) Punctuation; standard punctuation marks are used, i.e. . ; , ? ! - with one exception, i.e. quoted speech in the examples is indicated by left and right arrows ⟨...⟩ .

III. English equivalents: The following symbols occur in English equivalents and are to be interpreted thus:

a) A semicolon ; separates alternate English equivalents, e.g. Отвести' ду'шу To get something off one's chest; To unburden oneself.

b) A slash / indicates mutually exclusive elements, e.g. Чужи'ми рука'ми жар загреба'ть To make a cat's-paw/a dupe of someone.

c) Brackets [...] when an explanation rather than an equivalent is given for a specific idiom, it is placed in brackets, e.g. (Всегда') винова'т стре'лочник [Refers to situations in which an underling is blamed and becomes the scapegoat.] Brackets may also enclose a comment used to specify the context in which an idiom is used; e.g. Есть про'сит Ragged; worn-out; in need of repair [about clothing].

d) Parentheses ( ) signal "replacive" elements, e.g. туда' и доро'га [кому] It serves (him) right.

TABLE D

```
         TITLE 'IDIOM UPDATER'
UPDATER  CSECT
         SAVE  (14,12)
         BALR  12,0
         USING *,12
         LR    15,13
         L     13,=A(SAVEAREA)
         ST    13,8(15)
         ST    15,4(13)
         USING INAR,10
         USING OUTAR,11
         USING SAVEAREA,13
         MVI   LISW,0                  NO ELEMENT LISTING
         L     2,0(1)
         CLC   =H'3',0(2)
         BNE   OPEN
         CLC   =C'ALL',2(2)
         BNE   OPEN
         MVI   LISW,X'80'              ELEMENT LISTING OF ALL RECORDS
OPEN     OPEN  (IN,,OUT,OUTPUT,UPDT,,LIST,OUTPUT,MESS,OUTPUT)
         PUT   MESS,HDLINE
         LA    10,DREC1                PROVIDES INITIAL BLANKS FOR 'INID'
         L     15,=A(GETUP)            INITIALIZE UPDATE AREA
         BALR  14,15
         SPACE 2
*            GET INPUT RECORD
         SPACE
GET      GET   IN
         LR    10,1                        ESTABLISH ADDRESSABILITY
         ST    1,INAD
         NI    DATSO,X'BF'
         L     2,NINP
         LA    2,1(2)
         C     2,=F'30'
         BNH   TSKIP
         FEOV  IN
TSKIP    EQU   *
         ST    2,NINP
         SPACE
*            COLLATE INPUT AND UPDATE FILES
         SPACE
COLL     CLC   INID,UPID2
         BE    GOT1                    UPDATE RECORD
         BL    NOUP                    RECORD UNCHANGED
         SPACE
*            RECORD INSERTED
         SPACE
         CLC   UPFLD1(5),=C'00001'
         BE    INSOK
NOINS    LA    1,=C'12'                HAS TO BE INSERTION
         LA    0,UPAR2
```

```
         L      15,=A(UERR)
         BALR   14,15
BINS     L      15,=A(GETUP)          GET NEXT UPDATE
         BALR   14,15
         B      ENUP
INSOK    MVC    FREEL,=A(SUPLIST)     INITIALIZE AUX. LIST
         MVC    FREED,=A(SAVDAT)      INITIALIZE AUX. DATA AREA
         LA     1,UPIDM
         L      15,=A(GETDAT)         FIND EXTENT OF IDIOM
         BALR   14,15
         LTR    15,15
         BNZ    BINS                  SUB-ROUTINE ERROR RETURN
         ST     1,SUBCH               SAVE FINAL BYTE FOR PHASE1
         PUT    OUT
         LR     11,1
         ST     1,OUTAD
         L      2,NOUT
         LA     2,1(2)
         ST     2,NOUT
UP2OUT   MVC    OUTID,UPID2
         MVC    OUTSTAN,UPSTAN
         MVC    OUTFM,UPFM
         LA     2,ELAD
         ST     2,0(2)                INITIALIZE ELLIST
         LA     1,4(2)
         ST     1,ELTAB               BEGIN ADDR
         OI     DATSO,X'80'
         MVC    INTCH,DATEL
         NI     INTCH,0
         MVI    SECN,0
         L      15,=A(PHASE1)         MAKE ELEMENT TABLE
         BALR   14,15
         MVC    ELTAB+4(4),ELAD       END ADDR
         MVI    8(4),0
         MVC    9(23,4),8(4)          MAKE ZERO ENTRIES
         OI     INSW+1,X'FC'          TURN ON INSERTION SWITCH
         B      ANUP                  CONTINUE
*             RETURN VIA SWITCH
INSMES   NI     INSW+1,X'CF'
         LA     0,=C'-INSERTION '
         LA     1,OUTID
         L      15,=A(PRUP2)          PRINT NEW RECORD
         BALR   14,15
*            CHECK FOR END-OF-UPDATES
ENUP     CLC    UPID2,=9C'9'
         BNE    COLL
         CLC    INID,=9C'9'           CHECK FOR END-OF-JOB
         BE     EOJ
         B      COLL
         SPACE
*            RECORD IS NOT UPDATED
```

```
         SPACE
NOUP     PUT    OUT
         LR     11,1
         ST     1,OUTAD
         L      2,NOUT
         A      2,1(2)
         ST     2,NOUT
         MVC    HAFWD,INLN
         LH     2,HAFWD
         C      2,MAXL
         BNH    *+8
         ST     2,MAXL
         LR     3,2
         A      2,TOTL
         ST     2,TOTL
         BCTR   3,0
         TM     LISW,X'80'
         BNZ    ALIST
*             DO NOT LIST RECORD
         LA     2,INLN
         LA     4,OUTLN
LUPUT    C      3,=F'256'
         BL     FIPUT
         MVC    0(256,4),0(2)        INPUT TO OUTPUT RECORD
         LA     2,256(2)
         LA     4,256(4)
         S      3,=F'256'
         B      LUPUT
FIPUT    EX     3,PUTX
         B      GET
PUTX     MVC    0(0,4),0(2)
*            RECONSTITUTE AND LIST RECORD
ALIST    L      15,=A(SEXEL)
         BALR   14,15                UNPACK RECORD
         MVC    OUTID,INID
         MVC    OUTSTAN,INSTAN
         MVC    OUTFM,INFM
         L      15,=A(RECON)
         BALR   14,15
         B      GET
         SPACE
*            RECORD TO BE UPDATED
         SPACE
COT1     CLC    UPFLD1(5),=C'00000'
         BNE    PUTN
*            DELETE ENTIRE RECORD
         MVC    TCD,UPAR2
         L      15,=A(GETUP)
         BALR   14,15
         CLC    INID,UPID2
         BL     DELOK
```

```
         LA    1,=C'13'              UPDATE OF DELETED RECORD
         LA    0,ICD
         L     15,=A(UERR)
         BALR  14,15
         B     GOT1
DELOK    LA    0,=C'-DELETION  '
         LA    1,INID
         L     15,=A(PRUP2)          PRINT RECORD
         BALR  14,15
         B     GET
*             MODIFICATION UPDATE
PUTN     PUT   OUT
         LR    11,1
         ST    1,OUTAD
         L     2,NOUT
         LA    2,1(2)
         ST    2,NOUT
         MVC   OUTID,INID
         MVC   OUTSTAN,INSTAN
         MVC   OUTFM,INFM
         L     15,=A(SEXEL)
         BALR  14,15                 UNPACK RECORD
         MVC   FREEL,=A(SUPLIST)     INITIALIZE AUX. LIST
         L     2,=A(SAVDAT)
         ST    2,FREED               INITIALIZE AUX. DATA AREA
         SPACE
*            LOOP FOR SUCCEEDING UPDATES
         SPACE
OPLUP    CLI   UPFLD1,C'C'
         BNE   SPECUP
         CLC   UPEL,=C'000'
         BNE   OPER
         LA    2,L'OPS1
DECOP1   LA    3,OPS1-1(2)
         CLC   UPOP,0(3)
         BE    GOOP1
         BCT   2,DECOP1
         B     OPER
GOOP1    SLL   2,2
         B     *(2)
         B     BADDEL
         B     BADINS
         B     REPALL
         B     NEWF
         B     STANR
         B     TTERM
         B     ANUP                  PRINT RECORD (= NO OP)
*            IMPROPER DELETION OF ENTIRE RECORD
BADDEL   LA    1,=C'13'
         LA    0,UPAR2
         L     15,=A(UERR)
```

```
         BALR  14,15
         B     ANUP
*             INCORRECT INSERTION OF ENTIRE ENTRY
BADINS   LA    1,=C'12'
         LA    0,UPAR2
         L     15,=A(UERR)
         BALR  14,15
         B     ANUP
*             REPLACE HEADWORD (LINE ZERO)
NEWF     MVC   OUTFM,UPDAT2
         B     ANUP
*             REPLACE STANDARD RECORD
STANR    MVC   OUTSTAN,UPDAT2
         B     ANUP
*             INDICATE TECHNICAL TERM
TTERM    MVC   OUTSTAN+9(1),UPDAT2
         B     ANUP
*             REPLACE ENTIRE ENTRY
REPALL   MVC   OUTFM,UPFM
         MVC   OUTSTAN,UPSTAN
         MVC   TABOF,=F'0'
         LA    1,UPIDM                 REPLACE IDICM
UPEND    L     15,=A(GETDAT)
         BALR  14,15                   SAVE DATA
         LTR   15,15
         BNZ   ANUP
         ST    1,SUBCH
         OI    DATSO,X'80'
         MVC   INTCH,DATEL
         NI    INTCH,0
         LA    4,ELAD
         L     5,0(4)
         LA    5,4(5)
         LA    6,ELTAB
         A     6,TABOF                 REG6 POINTS TO ELTAB ENTRY
         ST    5,0(6)                  BEGIN ADDR
         MVC   SECN,TABOF+3
         L     15,=A(PHASE1)
         BALR  14,15
         L     5,0(4)                  REG4 POINTS TO ELAD
         ST    5,4(6)                  END ADDR
         B     ANUP
*             UPDATE OF SECTION OR ELEMENT
SPECUP   CLC   UPEL,=C'000'
         BH    BEGOP
         BL    OPER
         CLI   UPFLD1,C'0'
         BNH   OPER
         CLI   UPFLD1,C'4'
         BH    OPER
         IC    1,UPFLD1
```

```
         N     1,=F'15'
         BCTR  1,0
         SLL   1,3
         ST    1,TABUF                 SAVE ELTAB OFFSET
         CLI   UPOP,C'R'               R = REPLACE SECTION
         BNE   DSEC
         LA    1,UPDAT2
         B     UPEND                   CONTINUE
DSEC     CLI   UPOP,C'D'               D = DELETE SECTION
         BNE   OPER
         SR    2,2
         ST    2,ELTAB(1)              ZERO SECTION
         ST    2,ELTAB+4(1)
         B     ANUP
*          UPDATE ON AN ELEMENT WITHIN A SECTION
BEGOP    CLI   UPFLD1,C'0'
         BNH   OPER
         CLI   UPFLD1,C'4'
         BH    OPER
         IC    1,UPFLD1
         N     1,=F'15'
         BCTR  1,0
         SLL   1,3
         L     3,=A(ELTAB)
         AR    3,1
         CLC   =F'0',0(3)
         BNE   VALSEC
         LA    1,=C'18'                ELEMENT UPDATE IN EMPTY SECTION
         LA    0,UPAR2
         L     15,=A(UERR)
         BALR  14,15
         B     ANUP
VALSEC   L     3,0(3)
         PACK  DEC,UPEL
         CVB   2,DEC
         BCTR  2,0
         SLL   2,2
         LA    2,4(3,2)
         ST    2,BINEL
         C     2,ELAD
         BNH   INOPS2
         LA    1,=C'16'                NO SUCH ELEMENT NO.
         LA    0,UPAR2
         L     15,=A(UERR)
         BALR  14,15
         B     ANUP
INOPS2   LA    2,L'OPS2
DECOP2   LA    3,OPS2-1(2)
         CLC   UPOP,0(3)
         BE    GOOP2
         BCT   2,DECOP2
```

```
         B       UPER                    NO SUCH OP CODE
GOOP2    SLL     2,2
         B       *(2)
         B       TRUNC                   DELETE FINAL ELEMENTS
         B       INSERT                  INSERT ELEMENT
         B       REPL                    REPLACE AN ELEMENT
         B       OBLIK                   OBLIQUE(*) TO CANONICAL(:) X-REF
         B       DELETE                  DELETE AN ELEMENT
*              INSERT IDIOM ELEMENT
INSERT   L       2,BINEL
         S       2,=F'4'
         L       3,FREEL
         LA      1,UPDAT2
         L       15,=A(GETDAT)
         BALR    14,15
         LTR     15,15
         BNZ     ANUP
         CLI     0(2),X'FC'              'FC' MEANS PRIOR INSERT SUB-CHAIN
         BE      MULTIN
         C       3,=A(SUPLIST+L'SUPLIST-16)
         BNH     INS1
         LA      1,=C'14'                INTERNAL LIST EXHAUSTED
         LA      0,UPAR2
         L       15,=A(UERR)
         BALR    14,15
         B       ANUP
INS1     MVC     4(4,3),0(2)             MOVE OLC TABLE ENTRY
         ST      3,0(2)
         MVI     0(2),X'FC'              FLAG OLC TABLE ENTRY
         LA      4,8(3)
         ST      4,0(3)
         MVC     0(4,4),=F'0'            END OF SUB-CHAIN
         MVC     4(4,4),DATEL            INSERTION ELEMENT POINTER
         LA      4,8(4)
         ST      4,FREEL                 UPDATE LIST POINTER
         B       ANUP
MULTIN   C       3,=A(SUPLIST+L'SUPLIST-8)
         BNH     QUEND
         LA      1,=C'14'                INTERNAL LIST EXHAUSTED
         LA      0,UPAR2
         L       15,=A(UERR)
         BALR    14,15
         B       ANUP
QUEND    L       2,0(2)
         CLC     0(4,2),=F'0'            SEEK END OF SUB-CHAIN
         BNE     QUEND
         ST      3,0(2)                  ADD ELEMENT TO CHAIN
         MVC     0(4,3),=F'0'
         MVC     4(4,3),DATEL
         LA      3,8(3)
         ST      3,FREEL                 UPDATE POINTER
```

```
         B       ANUP
*              DELETION OF IDIOM ELEMENT(S)
DELETE   L       2,BINEL
         CLC     =CL3' ',UPDAT2
         BNE     DELMEL                  MULTIPLE ENTRY DELETION
         CLI     0(2),X'FC'
         BE      JUP1
         MVI     0(2),X'FF'              FLAG TABLE ENTRY FOR DELETION
         B       ANUP
JUP1     L       2,0(2)                  GO TO START OF LIST
         MVI     4(2),X'FF'
         B       ANUP
DELMEL   CLC     =C'000',UPDAT2
         BNL     OPER
         CLC     UPEL,UPDAT2
         BNL     OPER                    BAD 2ND ELEMENT NO.
         PACK    DEC,UPDAT2(3)
         CVB     3,DEC
         BCTR    3,0
         SLL     3,2
         LA      3,ELLIST(3)
         C       3,ELAD
         BNH     STARP
         LA      1,=C'16'                NO SUCH ELEMENT NO.
         LA      0,UPAR2
         L       15,=A(UERR)
         BALR    14,15
         B       ANUP
STARP    CLI     0(2),X'FC'
         BE      JUP2
         MVI     0(2),X'FD'              FLAG MEANS BEGIN DELETING
         B       DELSP
JUP2     L       2,0(2)
         MVI     4(2),X'FD'
DELSP    CLI     0(3),X'FC'
         BE      JUP3
         MVI     0(3),X'FE'              FLAG MEANS STOP DELETING
         B       ANUP
JUP3     L       3,0(3)
         MVI     4(3),X'FE'
         B       ANUP
*              REPLACEMENT
REPL     L       2,BINEL
         LA      1,UPDAT2
         L       15,=A(GETDAT)
         BALR    14,15
         LTR     15,15
         BNZ     ANUP
         CLI     0(2),X'FC'
         BNE     REPIN
         L       2,0(2)
```

```
         MVC   4(4,2),DATEL
         B     ANUP
REPIN    MVC   0(4,2),DATEL          REPLACE THE ELEMENT
         B     ANUP
*            REPLACE A CROSS-REFERENCED WORD IN AN OBLIQUE CASE WITH
*            ITS CANONICAL FORM
OBLIK    L     2,BINEL
         LA    3,4(2)
         C     3,ELAD
         BNH   ELADOK
         LA    1,=C'16'              NOT ENOUGH ELEMENTS
         LA    0,UPAR2
         L     15,=A(UERR)
         BALR  14,15
         B     ANUP
ELADOK   L     4,FREEL
         LA    1,UPDAT2
         L     15,=A(GETDAT)
         BALR  14,15
         LTR   15,15
         BNZ   ANUP
         CLI   0(3),X'FC'            'FC' MEANS PREVIOUS INSERTION
         BE    INSOB
         C     4,=A(SUPLIST+L'SUPLIST-24)
         BNH   INS2
         LA    1,=C'14'              INTERNAL LIST EXHAUSTED
         LA    0,UPAR2
         L     15,=A(UERR)
         BALR  14,15
         B     ANUP
INS2     MVI   0(2),X'FF'            DELETE X-REF ASTERISK
         MVC   4(4,4),0(3)
         ST    4,0(3)
         MVI   0(3),X'FC'            FLAG FOR INSERTION
         MVC   12(4,4),=X'7A000000' COLON (X'7A') MEANS CANONICAL X-REF
         MVC   20(4,4),DATEL         CANONICAL FORM
         LA    2,8(4)
         ST    2,0(4)                MAKE LINK
         LA    2,8(2)
         ST    2,8(4)                MAKE LINK
         MVC   16(4,4),=F'0'         END OF CHAIN
         LA    4,24(4)
         ST    4,FREEL               UPDATE INTERNAL LIST POINTER
         B     ANUP
INSOB    C     4,=A(SUPLIST+L'SUPLIST-16)
         BNH   SKWEZ
         LA    1,=C'14'              INTERNAL LIST EXHAUSTED
         LA    0,UPAR2
         L     15,=A(UERR)
         BALR  14,15
         B     ANUP
```

```
SKWEZ    MVI   C(2),X'FF'          DELETE X-REF ASTERISK
         L     3,0(3)
         L     5,C(3)
         MVC   4(4,4),=X'7A0C0C0C' INSERT COLON
         MVC   12(4,4),DATEL       INSERT CANONICAL FORM
         LA    2,8(4)
         ST    4,0(3)              INSERT LINKS IN CHAIN
         ST    2,C(4)
         ST    5,8(4)
         LA    4,16(4)
         ST    4,FREEL             UPDATE INTERNAL LIST POINTER
         B     ANUP
*            CUT OFF FINAL ELEMENTS
TRUNC    L     2,8INEL
         CLI   4(2),X'FC'
         BE    JUP4
         MVI   4(2),X'FD'
         B     ANUP
JUP4     L     2,4(2)
         MVI   4(2),X'FD'          FLAG MEANS BEGIN DELETING
         B     ANUP
*            UPDATE OPERAND FORMAT ERROR
OPER     LA    1,=C'17'
         LA    0,UPAR2
         L     15,=A(UERR)
         BALR  14,15
         SPACE
*            FINISHED WITH CURRENT UPDATE
         SPACE
ANUP     L     15,=A(GETUP)        GET THE NEXT UPDATE
         BALR  14,15
         CLC   OUTID,UPID2
         BE    OPLUP               UPDATE SAME RECORD AGAIN
*            NO MORE UPDATES, RECONSTITUTE RECORD
         L     15,=A(RECON)
         BALR  14,15
         MVC   HAFWD,OUTLN
         LH    1,HAFWD
         C     1,MAXL
         BNH   *+8
         ST    1,MAXL
         A     1,TOTL
         ST    1,TOTL
         SPACE
*            INSERTION SWITCH: NOP - OFF; B - ON.
INSW     NOP   INSMES
         SPACE
*            PRINT RECORD BEFORE AND AFTER UPDATING
         LA    0,=C'-OLD RECORD'
         LA    1,INID
         L     15,=A(PRUP2)        BEFORE
```

```
         BALR  14,15
         LA    0,=C'ONEW RECORD'
         LA    1,OUTID
         L     15,=A(PRUP2)          AFTER
         BALR  14,15
         B     GET
         SPACE
*             END-OF-INPUT ROUTINE
         SPACE
EOF      CLC   UPID2,=9C'9'
         BE    EOJ
         LA    10,=9C'9'
         S     10,=F'4'              SETS 'INID' TO ALL 9'S
         B     CELL
*             END OF RUN
EOJ      L     2,MAXL                WRITE STATISTICS MESSAGE
         CVD   2,DEC
         UNPK  STMAX,DEC
         OI    STMAX+2,C'0'
         LA    2,0
         L     3,TOTL
         D     2,NOUT
         CVD   3,DEC
         UNPK  STAVE,DEC
         OI    STAVE+2,C'0'
         L     2,NINP
         CVD   2,DEC
         UNPK  STIN,DEC
         OI    STIN+4,C'0'
         L     2,NOUT
         CVD   2,DEC
         UNPK  STOUT,DEC
         OI    STOUT+4,C'0'
         L     2,ERREC
         CVD   2,DEC
         UNPK  STERC,DEC
         OI    STERC+3,C'0'
         PUT   MESS,STATLN
*             END OF JOB
         CLOSE (IN,,OUT,,UPDT,,MESS,,LIST)
         STOP
         SPACE
MAXL     DC    F'0'                  MAX. RECORD LENTH
TOTL     DC    F'0'                  TOTAL LENGTH OF ALL OUTPUT RECORDS
NINP     DC    F'0'                  TOTAL INPUT RECORDS
NOUT     DC    F'0'                  TOTAL OUTPUT RECORDS
         SPACE
FREEL    DS    F                     ADDR OF AVAILABLE SPACE IN 'SUPLIST'
SUPLIST  DS    CL(18*8)              UPDATE ELEMENT TABLE ENTRIES (2F)
         SPACE
TCD      DS    CL80                  TO PASS UPDATE CARD IMAGE
```

```
         SPACE
DREC1    DC    H'13,C',CL9' '          INITIAL DUMMY RECORDS
         SPACE
OPS1     DC    C'DIRESKP'              OP CODE TABLE FOR ENTRY UPDATES
OPS2     DC    C'IIRCD'                OP CODE TABLE FOR ELEMENT UPDATES
         SPACE
STATLN   DC    C'1MAX='
STMAX    DC    C'000'                  MAXIMUM OUTPUT RECORD LENGTH
         DC  * C', AVE='
STAVE    DC    C'000'                  AVERAGE OUTPUT RECORD LENGTH
         DC    C', IN='
STIN     DC    C'00000'                NO. OF INPUT RECORDS
         DC    C', OUT='
STOUT    DC    C'00000'                NO. OF OUTPUT RECORDS
         DC    C', ERR='
STERC    DC    C'0000'                 INPUT ERROR RECORDS
         DC    CL93'.'
         LTORG
         TITLE '*** 1 ***  SET UP IDIOM ELEMENT TABLE (ELLIST)'
PHASE1   CSECT
         USING *,15
         STM   14,6,ELSAV
         LR    6,15
         DROP  15
         USING PHASE1,6
         USING DUMGP,5
         L     1,INTCH
         MVC   TP3(2),=XL2'0'          INITIALIZE FLAGS
         L     2,ELAD
         S     2,=F'4'
         ST    2,TESTAD                INITIALIZE DECREMENT ADDR
         OI    EOS2+1,X'F0'
         SR    2,2
         IC    2,SECN
         SRL   2,1
         L     2,LISTAB(2)
         ST    2,TABAD                 SAVE ADDR OF CURRENT TABLE
         LA    2,0
         LA    5,GP1                   INITIALIZE STACK WITH REG5
         BCTR  1,0
         SPACE
*           BEGIN 1ST PHASE, NEXT CTL CHAR LOOP
*             1ST PHASE ESTABLISHES LIST OF TEXT AND SYMBOL
*             ELEMENTS
         SPACE
TROOP    LA    4,BRACC                 NORMAL EXIT SWITCH ADDRESS
         ST    4,SWAD
         LA    3,1(1)                  BEGINNING CHAR IN REG3
         L     4,ELAD
         S     4,TESTAD
         C     4,=F'800'
```

```
         BNH   STEL
         LA    0,=C'09'          MORE THAN 200 ELEMENTS
         L     15,=A(ERR)
         BALR  14,15
         B     PEND
STEL     STH   4,CUREL
         L     4,SUBCH
         SR    4,3
         C     4,=F'40'          SCAN NO MORE THAN 40 CHAR'S
         BL    *+8
         LA    4,39
         L     1,TABAD           TRANSLATE TABLE ADDR
         SPACE
         EX    4,TRT             LOCK FOR NEXT CTL CHAR, ADDR IN REG1
         SPACE
         BNZ   CC2
         LA    1,1(4,3)          40 CHAR'S OF TEXT IF ZERO COND CODE
         LA    4,PEND            END-OF-IDIOM ADDRESS
         C     1,SUBCH
         BH    FLID
         LA    0,=C'22'          TEXT ELEMENT TOO LONG
         L     15,=A(ERR)
         BALR  14,15
         LA    4,BIGX            CONTINUE-SCAN ADDRESS
FLID     ST    4,SWAD
         B     TEXIS
CC2      BNH   CHTEX
         C     1,SUBCH
         BL    CHTEX             SECTION SCAN NOT COMPLETED
         NI    EOS2+1,X'0F'      LAST BYTE IS CONTROL CHAR
CHTEX    CR    1,3
         BNH   NOTEX
TEXIS    CLC   CURGP,=H'0'       TEXT BEFORE CONTROL CHAR
         BE    MTEXT
         CLC   CURSUB,=H'0'
         BH    MTEXT
         MVC   CURSUB,CUREL      BEGIN SUB-GROUP
MTEXT    ST    3,TEXEL
         LR    4,1
         SR    4,3
         BCTR  4,0
         STC   4,TEXEL
CHTP3    CLI   TP3,0
         BNZ   MTYP3             PRECEDING TYPE 3 CHAR
         CLI   GP,X'95'
         BNE   SEPTEX
         LA    5,5(5)            POP UP GROUP DELIMITER, END GROUP
SEPTEX   L     4,ELAD            TEXT ELEMENT ONLY
         MVC   4(4,4),TEXEL
         LA    4,4(4)
         ST    4,ELAD
```

```
         B       EOS1
MTYP3    CLI     TP3,X'95'
         BNE     WTP3
         CLI     TEXEL,C
         BNZ     WTP3
         CLI     C(3),C'R'
         BNE     WTP3
         CLI     GP,0
         BE      MXRR
         LA      0,=C'02'                REC'D X-REF WITHIN A GROUP
         L       15,=A(ERR)
         BALR    14,15
MXRR     S       5,=F'5'                 PUSH DOWN GROUP DELIMITER
         MVI     GP,X'95'
         MVI     XREIN,C'R'              ENGLISH 'R' MEANS RECORDED X-REF
         MVC     CURGP,CUREL
         MVC     CURSUB,CUREL
WTP3     L       4,ELAD                  TEXT IS X-REF'D
         MVC     4(4,4),TEXEL            MAKE TEXT ELEMENT
         LA      4,4(4)
         ST      4,ELAD
         MVI     TP3,0
         CLI     GP,X'95'
         BNE     EOS1
         LA      5,5(5)                  POP UP GROUP DELIMITER, END GROUP
         B       EOS1
NOTEX    CLI     TP3,0
         BE      EOS1
         LA      0,=C'03'                TYPE III CHAR NOT FOLLOWED BY TEXT
         L       15,=A(ERR)
         BALR    14,15
EOS1     L       15,SWAD
         BR      15                      EXIT SWITCH
BIGX     BCTR    1,0
         B       TROOP                   CONTINUE SCAN
*            DEAL WITH CTL CHAR
         SPACE
BRACC    B       *(2)
         B       INVCH
         B       TP1CH
         B       SLASH
         B       DUAL
         B       LEFTD
         B       RITD
         B       TP3CH
         SPACE
INVCH    LA      0,=C'01'                INVALID CHARACTER
         L       15,=A(ERR)
         BALR    14,15
         SPACE
TP1CH    CLC     CURGP,=H'0'             TYPE I CTL CHAR
```

```
         BE    TP1EL
         CLC   CURSUB,=H'0'
         BH    TP1EL
         MVC   CURSUB,CUREL          BEGIN SUB-GROUP
TP1EL    L     4,ELAD
         MVC   4(4,4),=F'0'
         MVC   4(1,4),0(1)           MAKE CTL CHAR ELEMENT
         LA    4,4(4)
         ST    4,ELAD
         B     EOS2
         SPACE
SLASH    CLC   CURSUB,=H'0'
         BNE   ENSUB
         LA    0,=C'04'              '/' DOES NOT FOLLOW SUB-GROUP TEXT
         L     15,=A(ERR)
         BALR  14,15
         B     TP2EL
ENSUB    MVC   CURSUB,=H'0'          END SUB-GROUP
         B     TP2EL
         SPACE
DUAL     CLC   GP,0(1)
         BE    RITD
         SPACE
LEFTD    C     5,=A(BSTAK)
         BH    LFOK
         LA    0,=C'21'              TOO MANY LEFT DELIMITERS
         L     15,=A(ERR)
         BALR  14,15
         B     TP2EL
LFOK     CLC   CURGP,=H'0'
         BE    DOLEF
         CLC   CURSUB,=H'0'
         BH    DOLEF
         MVC   CURSUB,CUREL          BEGIN SUB-GROUP
DOLEF    S     5,=F'5'               PUSH DOWN DELIMITER
         MVC   GP,0(1)
         MVC   CURGP,CUREL           BEGIN GROUP
         MVC   CURSUB,=H'0'
         B     TP2EL
         SPACE
RITD     CLI   GP,0
         BNE   RUP
         LA    0,=C'20'              TOO MANY RIGHT DELIMITERS
         L     15,=A(ERR)
         BALR  14,15
         B     TP2EL
RUP      LA    2,3
FINR     LA    4,T2TAB1(2)
         CLC   0(1,1),0(4)
         BE    REDUN
         BCT   2,FINR
```

```
RFOUN    LA    4,T2TAB2(2)
         CLC   GP,0(4)
         BE    RTOK
         LA    0,=C'05'          RIGHT DELIMITER DOESN'T MATCH LEFT
         L     15,=A(ERR)
         BALR  14,15
RTOK     LA    5,5(5)            POP UP DELIMITER, END GROUP
         SPACE
TP2FL    L     4,ELAD
         MVC   4(4,4),=F'0'
         MVC   4(1,4),0(1)       MAKE  CTL CHAR ELEMENT
         LA    4,4(4)
         ST    4,ELAD
         B     EOS2
         SPACE
TP3CH    CLC   CURGP,=H'0'
         BE    SAV3
         CLC   CURSUB,=H'0'
         BH    SAV3
         MVC   CURSUB,CUREL      BEGIN SUB-GROUP
SAV3     L     4,ELAD
         MVC   4(4,4),=F'0'
         MVC   4(1,4),0(1)
         LA    4,4(4)
         ST    4,ELAD
         MVC   TP3,0(1)
EOS2     B     TROOP             END-OF-IDIOM SWITCH
         SPACE 2
*            FINAL CHECKS
         SPACE
PEND     CLI   GP,0
         BNE   FLPEN
         CLI   TP3,0
         BE    CHERR
FLPEN    LA    0,=C'06'          FLAGS PENDING AT END OF 1ST PHASE
         L     15,=A(ERR)
         BALR  14,15
CHERR    TM    DATSO,X'80'
         BNZ   PH1OUT
         CLI   XREFIN,C'R'
         BNE   NORR
         CLI   INSTAN,C'0'       REFERENCE VS. SOURCE
         BE    PH1OUT
         B     ER07
NORR     CLI   INSTAN,C'1'       SOURCE VS. REFERENCE
         BE    PH1OUT
ER07     LA    0,=C'07'          DISCREPANCY WITH STANDARD RECORD
         L     15,=A(ERR)
         BALR  14,15
PH1OUT   LM    14,6,ELSAV
         RETURN
```

```
          SPACE
TRT       TRT   0(0,3),0(1)          SEEK CTL CHAR
          SPACE
TESTAD    DS    F                    DECREMENT ADDR IN ELLIST
TEXEL     DS    F                    TEMP. TEXT ELEMENT
SWAD      DS    F                    EXIT SWITCH ADDRESS
ELSAV     DC    9F'0'                REG SAVE AREA
*              TABLE OF TRANSLATE TABLE ADDRESSES
LISTAB    DC    A(TRTAB1)            SEC. 1
          DC    A(TRTAB3)            SEC. 2
          DC    A(TRTAB3)            SEC. 3
          DC    A(TRTAB3)            SEC. 4
TABAD     DC    A(0)                 CURRENT SECTION
          SPACE
*TRANSLATE AND TEST VALUES IN REG 2:
*              0- TEXT
*              4- INVALID CHARACTER
*              8- TYPE I CTL CHAR, PUNCTUATION, NO SPECIAL FUNCTION
*              12-24 TYPE II CTL CHAR, GROUP DELIMITERS
*              12- SLASH, SUB-GROUP DIVIDER
*              16- RIGHT OR LEFT DELIMITER
*              20- LEFT DELIMITER
*              24- RIGHT DELIMITER
*              28- TYPE III CTL CHAR, FOLLOWING TEXT MUST BE DEALT WITH
*
*              TABLE FOR IDIOM
TRTAB1    DC    64X'04'              INVALID
          DC    X'08'                BLANK
          DC    10X'04'
          DC    X'00'                PERIOD
          DC    X'08'                LESS-THAN SIGN
          DC    X'14'                LEFT PARENTHESIS
          DC    X'08'                PLUS SIGN
          DC    X'1C'                LOGICAL OR
          DC    X'14'                AMPERSAND
          DC    9X'04'
          DC    X'00'                EXCLAMATION POINT
          DC    X'10'                DOLLAR SIGN
          DC    X'1C'                ASTERISK
          DC    X'18'                RIGHT PARENTHESIS
          DC    X'08'                SEMI-COLON
          DC    X'04'                LOGICAL NOT, (INVALID)
          DC    X'0C'                HYPHEN
          DC    X'0C'                SLASH
          DC    9X'04'
          DC    X'00'                COMMA
          DC    X'00'                PERCENT SIGN
          DC    X'1C'                UNDERSCORE
          DC    X'08'                GREATER-THAN SIGN
          DC    X'0C'                QUESTION MARK
          DC    10X'04'
```

```
         DC    X'1C'                COLON
         DC    X'04'                POUND SIGN,(INVALID)
         DC    X'18'                AT SIGN
         DC    X'00'                APOSTROPHE
         DC    X'1C'                EQUAL SIGN
         DC    X'00'                QUOTATION MARK
         DC    18X'04'
         DC    X'00'                X'92' - UPPER CASE RUSSIAN
         DC    2X'04'
         DC    X'1C'                X'95' - BEGIN ENGLISH
         DC    X'08'                X'96' - BEGIN UPPER CASE RUSSIAN
         DC    X'08'                X'97' - RESUME LOWER CASE RUSSIAN
         DC    41X'04'
         DC    9X'00'               A THRU I
         DC    7X'04'
         DC    7X'00'               J THRU P
         DC    X'00'                Q
         DC    X'00'                R
         DC    8X'04'
         DC    8X'00'               S THRU Z
         DC    6X'04'
         DC    2X'00'               0,1
         DC    7X'00'               2 THRU 8
         DC    X'00'                9
         DC    6X'04'
*          TABLE FOR EXAMPLE, EQUIV.
TRTAB3   DC    64X'04'
         DC    X'08'                BLANK
         DC    9X'04'
         DC    X'C800C814080814'    C' .<(+|&'
         DC    9X'04'
         DC    X'C0C808180808C0C8'  C' $*);¬-/'
         DC    9X'04'
         DC    X'C0C0C80800'        C',%_>?'
         DC    10X'04'
         DC    X'C808180C08C0'      C':#@'='"'
         DC    18X'04'
         DC    X'C0C4C4C80808'      X'92,(93,94),95,96,97'
         DC    26X'04'
         DC    X'C8C4C4C80808'      X'82,(83,84),85,86,87'
         DC    9X'04'
         DC    9X'00'               A THRU I
         DC    7X'04'
         DC    9X'00'               J THRU R
         DC    8X'04'
         DC    8X'00'               S THRU Z
         DC    6X'04'
         DC    10X'00'              0 THRU 9
         DC    6X'04'
         SPACE
CUREL    DS    H                    CURRENT ELEMENT NO.
```

```
         SPACE
TP3      DS    C                    TYPE III CTL CHAR. FLAG
XREFIN   DS    C                    FLAGS PRESENCE OF REC'D X-REF
         SPACE
T2TAB1   DC    C')@=$'              RIGHT DELIMITERS DECODER
         DC    X'95'
T2TAB2   DC    C'(&&=$'             LEFT DELIMITERS DECODER
         DC    X'95'
         SPACE
BSTAK    DS    5XL5                 GROUP DELIMITER STACK - 5 BYTE ELEM.
GP1      DC    XL5'0'
*          "TEMPLATE" FOR GROUP DELIMITER STACK
DUMGP    DSECT
GP       DS    C                    GROUP DELIMITER FLAG
CURGP    DS    CL2                  GROUP DELIMITER ELEMENT NO.
CURSUB   DS    CL2                  SUB-GROUP DELIMITER ELEMENT NO.
PHASE1   CSECT
         LTORG
         TITLE '*** 2 ***  GET AN UPDATE RECORD'
GETUP    CSECT
         USING *,15
         STM   14,4,GETSAV
         LR    4,15
         DROP  15
         USING GETUP,4
         MVC   GETSAV+4,=F'0'
         NI    DATSO,X'DF'
UP1SW    B     GETUP1               FIRST-TIME SWITCH
         MVC   UPAR2(15),UPAR1      MOVE FOLLOWING UPDATE TO CURRENT
         LA    2,UPAR2+15
         CLC   UPID2,=9C'9'
         BE    UPOUT                END-OF-UPDATES
STICON   MVC   0(65,2),UPDAT1       MOVE UPDATE DATA
         LA    2,65(2)              NEXT POSITION IN REG2
AGETUP   GET   UPDT,UPAR1           GET FOLLOWING UPDATE
         CLC   SUPID,UPID1
         BNH   UPSAV
         LA    1,=C'10'             UPDATE SEQUENCE ERROR
         LA    0,UPAR1
         L     15,=A(UERR)
         BALR  14,15
         B     AGETUP
UPSAV    MVC   SUPID,UPID1          SAVE FOLLOWING UPDATE ID
         CLC   UPID1,UPID2
         BNE   UPOUT
         CLC   UPCON,=CL4' '
         BNE   UPOUT
*          CONTINUATION CARD
         C     2,=A(CLIM)
         BL    STICON
         LA    1,=C'11'             TOO MANY CONTINUATIONS
```

```
         LA    0,UPAR1
         L     15,=A(UERR)
         BALR  14,15
         B     ACETUP
UPOUT    BCTR  2,0
         ST    2,ULIM
         LM    14,4,GETSAV
         RETURN
         SPACE
*             INITIALIZE CARD AREAS
GETUP1   NI    UPISW+1,X'0F'
         GET   UPDT,UPAR2
         LA    2,UPAR2+80
         B     ACETUP
*            LAST UPDATE FROM SYSTEM
EOU      MVC   UPID1,=9C'9'
         B     UPOUT
         SPACE
GETSAV   DC    7F'0'                  SUB-ROUTINE SAVE AREA
*            FOLLOWING UPDATE AREA
UPAR1    DS    0CL80
UPID1    DC    C'999999999'           DEFAULT ID IS HIGH
UPCNT1   DC    C' '                   CONTINUATION COUNT
UPCON    DC    CL4' '                 BLANK IF CONTINUATION
UPOP1    DC    C' '                   UPDATE OP CODE
UPDAT1   DC    CL65' '
         SPACE
SUPID    DC    CL9' '                 UPDATE SAVE ID
         LTORG
         TITLE '*** 3 ***  SAVE UPDATE DATA'
GETDAT   CSECT
         USING *,15
         STM   14,5,GDSAV
         LR    5,15
         DROP  15
         USING GETDAT,5
         MVC   GDSAV+4,=F'0'          NORMAL RETURN CODE
         L     2,ULIM
         SR    2,1
DUPLUP   LA    3,0(1,2)
         CLI   0(3),C' '              FIND LAST NON-BLANK
         BNE   *+8
         BCT   2,DUPLUP
         L     3,FREED
         LA    4,1(2,3)
         C     4,=A(SAVDAT+L'SAVDAT)
         BNH   XSAVM
         LA    1,=C'15'               NO MORE SPACE IN SAVE AREA
         LA    0,UPAR2
         L     15,=A(UERR)
         BALR  14,15
```

```
         MVC   GDSAV+4,=F'4'        ERROR RETURN CODE
         B     GDOUT
XSAVM    ST    4,FREED
         BCTR  4,0
         ST    4,GDSAV+12           RETURN IN REG1
         ST    3,DATEL              ADDRESS
         STC   2,DATEL              LENGTH
         C     2,=F'64'
         BL    *+8
         MVI   DATEL,63             MAKE LESS THAN 65 CHAR'S.
TESL     C     2,=F'256'
         BNL   FULM
         EX    2,SAVMOV             MOVE TO SAVE AREA
GDOUT    LM    14,5,GDSAV
         RETURN
FULM     MVC   0(256,3),0(1)
         LA    3,256(3)
         LA    1,256(1)
         S     2,=F'256'
         B     TESL
         SPACE
SAVMOV   MVC   0(0,3),0(1)
         SPACE
GDSAV    DC    8F'0'                REGISTER SAVE AREA
         LTORG
DATSEC   CSECT
SAVDAT   DS    CL1024               AREA FOR SAVING DATA
         TITLE '*** 6 ***  WRITE ERROR MESSAGE'
ERR      CSECT
         USING *,15
         STM   14,7,ESV
         LR    7,15
         DROP  15
         USING ERR,7
         L     2,ERREC
         LA    2,1(2)
         ST    2,ERREC
         LR    2,0
         MVC   ERNO,0(2)
         LR    3,1                  GET POSITION IN IDIOM WHEN ERROR
         L     2,INTCH              TURNS UP
         BCTR  2,0
         SR    3,2
         CVD   3,DEC
         UNPK  ERPOS,DEC
         OI    ERPOS+2,C'0'
         NI    DUPSW+1,X'0F'
         TM    DATSO,X'80'
         BZ    WRIN
         MVC   ERID(116),ERID-1
         MVC   ERID,UPID2
```

```
         TM    DATSO,X'20'
         BZ    *+8
         OI    DUPSW+1,X'F0'        DATA ALREADY WRITTEN
         OI    DATSO,X'20'
         B     SAMP
WRIN     MVC   ERID,INID
         MVC   ERSTAN,INSTAN
         MVC   ERFM,INFM
         TM    DATSO,X'40'
         BZ    *+8
         OI    DUPSW+1,X'F0'        DATA ALREADY WRITTEN
         OI    DATSO,X'40'
SAMP     PUT   MESS,ERLINE
         AP    NOLN,=P'03'
DUPSW    NOP   EROUT
*             WRITE IDIOM IN HEXADECIMAL
         OI    MESW+1,X'F0'
         L     2,INTCH
         MVC   HXLIN0(2),=C'0 '
MESLN    LA    3,1
         MVC   HXLIN0+2(256),HXLIN0+1      BLANK PRINT LINES
         MVC   HXLIN0+258(141),HXLIN0+257
LTEST1   C     2,SUBCH
         BNH   LTEST2                 FINISHED IF HIGH
         NI    MESW+1,X'0F'
         B     RIMES
LTEST2   C     3,=F'100'
         BH    RIMES                  PRINT LINES FILLED IF HIGH
         LA    4,HXLIN0+5(3)
         MVC   0(1,4),0(2)            SAVE CHARACTER IN FIRST LINE
         LA    5,0
         IC    5,0(2)
         LR    6,5
         N     6,=F'15'               2ND HEX DIGIT IN REG6
         SRL   5,4                    1ST HEX DIGIT IN REG5
         LA    4,HXLIN1+5(3)
         IC    5,HEXTAB(5)
         STC   5,0(4)                 STORE 1ST HEX CHARACTER
         LA    4,HXLIN2+5(3)
         IC    6,HEXTAB(6)
         STC   6,0(4)                 STORE 2ND HEX CHARACTER
         LA    2,1(2)
         LA    3,1(3)
         B     LTEST1
RIMES    PUT   MESS,HXLIN0            WRITE IDIOM AND HEX IMAGES
         PUT   MESS,HXLIN1
         PUT   MESS,HXLIN2
         AP    NOLN,=P'04'
         CP    NOLN,=P'56'
         BNH   MESW
         PUT   MESS,HDLINE
```

```
         ZAP   NOLN,=P'01'
MESW     B     MESLN                    END-OF-OPERATION SWITCH
EROUT    LM    14,7,ESV
         RETURN
         SPACE
ESV      DC    10F'0'                   ERROR ROUTINE SAVE AREA
         SPACE
ERLINE   DC    C'-'                     ERROR MESSAGE
         DC    C' '
ERNO     DC    C'00'
         DC    CL6' '
ERPOS    DC    C'000'
         DC    CL4' '
ERID     DC    C'0000000000'
         DC    CL3' '
ERSTAN   DC    C'000000000000'
         DC    CL4' '
ERFM     DC    CL30' '
         DC    CL58' '
HXLIN0   DS    CL133                    IDIOM IMAGE
HXLIN1   DS    CL133                    FIRST HEX DIGITS
HXLIN2   DS    CL133                    2ND HEX DIGITS
HEXTAB   DC    C'0123456789ABCDEF'      HEX TRANSLATE TABLE
         LTORG
         TITLE '*** 7 ***  WRITE UPDATE ERROR MESSAGE'
UERR     CSECT
         USING *,15
         STM   14,3,URSV
         LR    3,15
         DROP  15
         USING UERR,3
         L     2,ERREC
         LA    2,1(2)
         ST    2,ERREC
         MVC   UERNO,0(1)
         LR    1,0
         MVC   ECI,0(1)
         PUT   MESS,ERMES
         AP    NOLN,=P'02'
         CP    NOLN,=P'56'
         BNH   UOUT
         PUT   MESS,HOLINE
         ZAP   NOLN,=P'01'
UOUT     LM    14,3,URSV
         RETURN
         SPACE
URSV     DC    6F'0'
         SPACE
ERMES    DC    C'0'
UERNO    DC    CL2' '
         DC    C' /'
```

```
ECI      DC    CL80' '
         DC    CL48'/'
         LTORG
         TITLE '*** 8 ***  PRINT UPDATED RECORD (VERSION 2)'
PROP2    CSECT
         USING *,15
         STM   14,5,PRSAV2
         LR    5,15
         DROP  15
         USING PROP2,5
         LR    2,0
         MVC   UMOPT,0(2)
         MVC   UMID,0(1)
         MVC   UMSTAN,INSTAN-INID(1)
         MVC   UMFM,INFM-INID(1)
         AP    NOLN,=P'02'
         CLI   UMOPT,C'0'
         BE    *+10
         AP    NOLN,=P'01'
         MVC   CHTAB,=X'B7B5B7B5'  INITIALIZE CTL CHAR TABLE
         LA    3,INDAT-INID(1)
         MVC   HAFWD,0(3)
         LH    4,HAFWD
         BCTR  4,0                 LENGTH OF DATA IN REG4
         LA    3,2(3)              ADDR OF DATA IN REG3
         C     4,=F'67'
         BL    ALIDM
         MVC   UMIDMA,0(3)
         PUT   MESS,UPMES1
         S     4,=F'67'
         LA    3,67(3)             PRINT MORE IDIOM (1ST SECTION)
         MVC   URC,CHTAB
         B     NEXEG
ALIDM    MVI   UMIDMA,C' '
         MVC   UMIDMA+1(L'UMIDMA-1),UMIDMA
         EX    4,PMOV1
         PUT   MESS,UPMES1         FINISHED WITH IDIOM
NEXUB    CP    NOLN,=P'56'
         BNH   SHTAB
         PUT   MESS,HDLINE
         ZAP   NOLN,=P'01'
SHTAB    MVC   CHTAB,CHTAB+1       NEXT CTL CHAR
         CLI   CHTAB,0
         BE    PROUT               ZERO MEANS FINISHED
         MVC   URC,CHTAB
         LA    3,1(4,3)
         MVC   HAFWD,0(3)
         LH    4,HAFWD
         BCTR  4,0
         LA    3,2(3)
         LTR   4,4
```

```
         BM    NEXOB                    ZERO SECTION LENGTH
NEXEG    C     4,=F'125'
         BL    FSEG
         MVC   UMIDMB,C(3)
         PUT   MESS,UPMES2
         AP    NOLN,=P'01'
         S     4,=F'125'
         LA    3,125(3)
         B     NEXEG                    PRINT MORE DATA FROM SECTION
FSEG     MVC   UMIDMB,UMIDMB-1
         EX    4,PMOV2
         PUT   MESS,UPMES2              ALL OF SECTION PRINTED
         AP    NOLN,=P'01'
         B     NEXOB                    NEXT SECTION
PROUT    LM    14,5,PRSAV2
         RETURN
         SPACE
PMOV1    MVC   UMIDMA(0),0(3)           MOVE TO INITIAL LINE
PMOV2    MVC   UMIDMB(0),0(3)           MOVE TO ADDITIONAL LINE
         SPACE
PRSAV2   DC    8F'0'
         SPACE
CHTAB    DS    CL4                      URP CTL CHAR TABLE
         DC    X'0C'                    FLAGS END OF TABLE
         SPACE
*              INITIAL MESSAGE LINE
UPMES1   DS    0CL133
UMOPT    DC    CL11'-'
         DC    C' '
UMID     DC    CL9' '
         DC    C' '
UMSTAN   DC    CL12' '
         DC    X'B6'
UMFM     DC    CL30' '
         DC    X'B7'
UMIDMA   DC    CL67' '
*              ADDITIONAL MESSAGE LINE
UPMES2   DC    CL6' '
URC      DC    X'B7'
         DC    C' '
UMIDMB   DC    CL125' '
         LTORG
         TITLE '*** 9 ***  SET UP SECTION ELEMENT TABLE'
SEXEL    CSECT
         USING *,15
         STM   14,7,SEXAV
         LR    7,15
         DROP  15
         USING SEXEL,7
         LA    2,ELAD
         ST    2,0(2)                   INITIALIZE ELLIST
```

```
         LA    2,INDAT
         LA    6,0
SUBREC   LA    3,1(2)
         CLC   =H'0',0(2)
         BNE   SUBDAT
         LA    4,0
         ST    4,ELTAB(6)
         ST    4,ELTAB+4(6)
         B     NESB
SUBDAT   MVC   HAFWD,0(2)
         LA    1,2(2)              BEGIN DATA IN REG1
         ST    1,INTCH
         AH    3,HAFWD             END DATA IN REG3
         ST    3,SUBCH
         L     5,ELAD              GET BEGIN ADDR (IN ELLIST)
         LA    5,4(5)
         ST    5,ELTAB(6)
         NI    DATSD,X'7F'
         STC   6,SECN
         L     15,=A(PHASE1)       FORM ELEMENT LIST
         BALR  14,15
         L     5,ELAD              GET END ADDR (IN ELLIST)
         ST    5,ELTAB+4(6)
NESB     LA    6,8(6)
         LA    2,1(3)
         C     6,=F'24'
         BNH   SUBREC
         LM    14,7,SEXAV
         RETURN
         SPACE
SEXAV    DC    10F'0'
         LTORG
         TITLE '*** IC ***  RECONSTITUTE RECORD FROM ELLIST'
RECON    CSECT
         USING *,15
         STM   14,12,RESAV
         LR    12,15
         DROP  15
         USING RECON,12
         MVC   PTAB,=F'0'
         LA    2,OUTDAT
         ST    2,DACT
         TM    LISW,X'80'
         BZ    SKIPR1
         CP    LISLN,=P'50'
         BL    LIN1
         PUT   LIST,PRHD
         ZAP   LISLN,=P'0'
LIN1     MVC   PRID,OUTID
         MVC   PRSTAN,OUTSTAN
         MVC   PREM,OUTEM
```

```
         PUT   LIST,PRO              WRITE 1ST LINE
         AP    LISLN,=P'2'
SKIPR1   LA    4,4
         LA    8,0
*             START MAKING UP A SECTION
DOSEC    LA    1,ELTAB
         A     1,PTAB
         SR    6,6
         CLC   =F'0',0(1)
         BNE   BEGEN
         L     2,DACT
         MVC   0(2,2),=H'0'          PUT ZERO SECTION LENGTH IN RECORD
         LA    2,2(2)
         ST    2,DACT
         B     NEXAB
BEGEN    TM    LISW,X'80'
         BZ    SKIPR2
         L     2,PTAB                PREPARE PRINT LINES
         SLL   2,1
         LA    2,SCNAM(2)
         MVC   SECID,0(2)
         MVC   PR1DT,PR1DT-1
         MVC   PR2DT,PR2DT-1
         MVC   PRDAT,PRDAT-1
SKIPR2   L     2,DACT
         LA    2,2(2)
         ST    2,OUPS
         L     3,0(1)
         L     5,4(1)
         ZAP   ELCO1,=P'0'
         OI    PUTUP+1,X'F0'
*             BEGIN ADDING ELEMENTS
PUTUP    B     NODUP                 MULTIPLE DELETE SWITCH
         CLI   0(3),X'FC'
         BE    STUP1
         CLI   0(3),X'FE'
         BNE   NEXUP
         OI    PUTUP+1,X'F0'         END OF DELETES, SWITCH OFF
         B     NEXUP
STUP1    L     2,0(3)                FIRST LINK OF SUB-CHAIN
         L     7,OUPS                OUTPUT ADDR
         CLI   4(2),X'FE'
         BNE   NEXUP
         OI    PUTUP+1,X'FC'         'FE' TURNS SWITCH OFF
         B     NEXLUP
*            MULTIPLE DELETES NOT IN EFFECT
NODUP    CLI   0(3),X'FC'
         BNE   PUPEL
         L     2,0(3)
         L     7,OUPS                OUTPUT ADDR
         CLI   4(2),X'FD'
```

```
         BNE   DELSG1
         NI    PUTUP+1,X'0F'          BEGIN MULTIPLE DELETES, SWITCH ON
         B     NEXUP
DELSG1   CLI   4(2),X'FF'
         BE    NEXLUP                 DELETE ONE ELEMENT
LISLUP   CLC   =XL3'0',5(2)           SUB-CHAIN LOOP
         BNE   UPTEX1
         MVC   0(1,7),4(2)            MOVE CONTROL CHARACTER TO OUTPUT
         LA    7,1(7)
         B     NEXLUP
UPTEX1   IC    8,4(2)
         L     9,4(2)
         EX    8,MOVIT                MOVE TEXT TO OUTPUT RECORD
         LA    7,1(8,7)
NEXLUP   CLC   0(4,2),=F'0'           ZERO MEANS END OF SUB-CHAIN
         BE    SAVAD
         L     2,0(2)                 NEXT LINK IN SUB-CHAIN
         B     LISLUP
*             NOT IN INSERTION SUB-CHAIN
PUPEL    CLI   0(3),X'FD'
         BNE   DELSG2
         NI    PUTUP+1,X'0F'          'FD' MEANS BEGIN MULTIPLE DELETES
         B     NEXUP
DELSG2   CLI   0(3),X'FF'
         BE    NEXUP                  DELETE ONE ELEMENT
         AP    ELCO1,=P'1'
         CLC   =XL3'0',1(3)
         BNE   UPTEX2
         L     7,OUPS
         MVC   0(1,7),0(3)            MOVE CTL CHAR
         TM    LISw,X'80'
         BZ    SKIPR3
         LA    7,PRDAT(6)
         MVC   0(1,7),0(3)
SKIPR3   LA    8,0
         TM    LISw,X'80'
         BZ    INCAD
         B     PRNEX
UPTEX2   IC    8,0(3)
         L     9,0(3)
         L     7,OUPS
         EX    8,MOVIT                MOVE TEXT
         TM    LISw,X'80'
         BZ    INCAD
         LA    7,PRDAT(6)             PRINT TEXT
         EX    8,MOVIT
         UNPK  ELCO2,ELCO1
         OI    ELCO2+1,X'F0'
         LA    7,PR1DT(6)             PRINT 1ST HEX DIGIT
         MVC   0(1,7),ELCO2
         LA    7,PR2DT(6)             PRINT 2ND HEX DIGIT
```

```
         MVC   0(1,7),ELCO2+1
PRNEX    LA    6,1(8,6)
         C     6,=F'121'          CHECK FOR PRINT-LINE OVERFLOW
         BL    INCAD
         CP    LISLN,=P'56'
         BL    LIN3A
         PUT   LIST,PRHD
         ZAP   LISLN,=P'0'
LIN3A    PUT   LIST,PR1           WRITE LINES
         PUT   LIST,PR2
         PUT   LIST,MS
         AP    LISLN,=P'3'
         S     6,=F'121'
         MVC   PR1DT,PR1DT-1
         MVC   PR2DT,PR2DT-1      CLEAR PRINT LINES
         MVC   PRDAT,PRDAT-1
         MVC   PR1DT(1),PR1DT+120 START NEW LINES
         MVC   PR2DT(1),PR2DT+120
         EX    5,SHIF
         LA    6,1(6)
INCAD    L     7,OUPS
         LA    7,1(8,7)           NEXT DATA POSITION
SAVAD    ST    7,OUPS
NEXUP    BXLE  3,4,PUTUP
*             FALL-THRU WHEN ALL ELEMENTS STORED IN OUTPUT RECORD
         L     2,DACT
         LA    1,2(2)
         L     3,OUPS
         ST    3,DACT
         SR    3,1
         STH   3,HAFWD
         MVC   0(2,2),HAFWD       STORE SECTION LENGTH
*            FINISH WRITING PRINT LINES
         TM    LISW,X'80'
         BZ    NEXAB
         PUT   LIST,PR1
         PUT   LIST,PR2
         PUT   LIST,MS
         AP    LISLN,=P'3'
         CP    LISLN,=P'56'
         BNH   NEXAB
         PUT   LIST,PRHD
         ZAP   LISLN,=P'0'
*CHECK FOR END OF TABLE OF SECTION ELEMENTS
NEXAB    L     1,PTAB
         LA    1,8(1)
         ST    1,PTAB
         C     1,=F'24'
         BNH   DOSEC
FINP     L     2,DACT
         LR    3,2
```

```
         BCTR  3,0                      LAST BYTE ADDR IN REG3
         LA    1,OUTLN
         SR    2,1
         STH   2,HALFWD
         MVC   OUTLN,HALFWD
         LM    14,2,RESAV
         LM    4,12,RESAV+24
         RETURN
         SPACE
MOVIT    MVC   0(0,7),0(9)
SHIF     MVC   PRDAT(0),PRDAT+120
         SPACE
RESAV    DC    15F'0'                   REG SAVE AREA
DACT     DS    F                        ADDR OF SECTION
OUPS     DS    F                        OUTPUT DATA POSITION
PTAB     DS    F                        OFFSET IN ELTAB
         SPACE
ELCO1    DS    PL2                      ELEMENT COUNTER (PACKED)
ELCO2    DS    CL2                      ELEMENT COUNTER (ZONED)
*            FOUR-LINE MESSAGE GROUP FOR LIST FILE
*            1ST LINE - ONE PER OUTPUT RECORD
PRO      DC    C'0'
PRID     DC    CL9' '
         DC    C' '
PRSTAN   DC    CL12' '
         DC    X'B6'
PREM     DC    CL30' '
         DC    CL79' '
*            2ND LINE - 1ST DIGITS OF ELEMENT NOS.
PR1      DC    CL13' '
PR1DT    DC    CL120' '
         DC    C' '                     1ST OVERFLOW BYTE
*            3RD LINE - 2ND DIGITS OF ELEMENT NOS.
PR2      DC    CL13' '
PR2DT    DC    CL120' '
         DC    C' '                     1ST OVERFLOW BYTE
*            4TH LINE - CONTAINS DATA FROM RECORD
MS       DC    C' '
SECID    DC    CL11' '                  SECTION IDENTIFIER AND URP CHAR.
         DC    C' '
PRDAT    DC    CL120' '
         DC    CL64' '                  OVERFLOW AREA
*            HEADING LINE
PRHD     DC    CL47'1'
         DC    C'LIST OF RECORDS, INCLUDING ELEMENT NOS.'
         DC    CL47' '
*            LINE COUNT FOR LIST FILE
LISLN    DC    PL2'60'
*            TABLE OF SECTION IDENTIFIERS
SCNAM    DS    0CL16                    ENTRY LENGTH IS 16 BYTES
         DC    CL5' '
```

```
         DC    C'IDIOM'
         DC    X'87'
         DC    CL5' '
         DC    C'EQUIVALENT'
         DC    X'85'
         DC    CL5' '
         DC    CL2' '
         DC    C'EXAMPLES'
         DC    X'87'
         DC    CL5' '
         DC    CL4' '
         DC    C'SYNTAX'
         DC    X'85'
         DC    CL5' '
         LTORG
         TITLE 'GENERAL CONSTANTS AND AREAS'
SAVEAREA CSECT
         DC    18F'0'                  SAVE AREA
DEC      DS    D                       DECIMAL CONVERSION
INAD     DS    F                       INPUT RECORD ADDRESS
OUTAD    DS    F                       OUTPUT RECORD ADDRESS
BINEL    DS    F                       TABLE OFFSET VALUE OF ELEMENT NO.
DATEL    DS    F                       LENGTH + ADDR OF DATA
TABOF    DS    F                       OFFSET FOR ELTAB (SECTION ELEMENTS)
ERREC    DC    F'0'                    ERROR RECORDS
ULIM     DS    F                       ADDR OF END OF LAST CARD
FREED    DS    F                       ADDR OF AVAILABLE SPACE IN 'SAVDAT'
*            PARAMETERS FOR UNPACKING DATA
DATSO    DC    XL1'0'                  BIT 0-SOURCE; 1-RECORD; 2-UPDATE
SECN     DC    XL1'0'                  SECTION NO. OFFSET (-1X8)
         DC    XL2'0'
INTCH    DS    F                       INITIAL PHASE1 POSITION
SUBCH    DS    F                       ADDR OF LAST BYTE
*            BEGIN AND END ADDR IN ELLIST FOR ELEMENTS OF EACH SECTION
ELTAB    DS    2F                      SEC1- IDIOM
         DS    2F                      SEC2- ENGLISH EQUIVALENT
         DS    2F                      SEC3- EXAMPLES
         DS    2F                      SEC4- SYNTACTIC ROLE (NOT ADDED)
         SPACE
ELAD     DS    F                       POINTER TO END OF ELLIST
ELLIST   DS    32CF                    STRING OF TEXT OR SYMBOL ELEMENTS
         SPACE
HAFWD    DC    F'0'                    PROVIDES A HALF-WORD BOUNDARY + ZERO
*            CURRENT UPDATE AREA
UPAR2    DS    0CL80
UPID2    DC    C'999999999'            DEFAULT ID IS HIGH
UPCNT2   DC    C' '                    CONTINUATION
UPFLD1   DC    C' '                    RECORD SECTION NO.
UPEL     DC    CL3' '                  UPDATED ELEMENT NO.
UPOP.    DC    C' '                    UPDATE OP CODE
UPDAT2   DS    CL585                   UPDATE DATA FOR REPLACE, INSERT, ETC
```

```
UPFM     DC    CL30' '              LINE ZERO (NEW RECORD ONLY)
UPSTAN   DC    CL12' '              STANDARD RECORD (NEW RECORD ONLY)
UPIDM    DC    CL23' '              IDIOM (NEW RECORD ONLY)
         DS    CL520                EXTRA SPACE FOR CONTINUATION
CLIM     EQU   *                    CONTINUATION LIMIT
*              INPUT RECORD AREA
INAR     DSECT
INLN     DS    CL4                  VARIABLE RECORD LENGTH
INID     DS    CL9                  ID
INSTAN   DS    CL12                 STANDARD RECORD
INFM     DS    CL30                 LINE ZERO
INDAT    DS    CL1                  BEGINNING OF DATA
*              OUTPUT RECORD AREA
OUTAR    DSECT
OUTLN    DS    CL4                  VARIABLE RECORD LENGTH
OUTID    DS    CL9                  ID
OUTSTAN  DS    CL12                 STANDARD RECORD
OUTFM    DS    CL30                 LINE ZERO
OUTDAT   DS    CL1                  BEGINNING OF IDIOM, ETC.
SAVEAREA CSECT
         SPACE
NCLN     DC    P'01'                LINE COUNT FOR 'MESS' FILE
HDLINE   DC    CL40'1'
         DC    C'IDIOM UPDATE MESSAGES'
         DC    CL72' '
         SPACE
LISW     DC    X'00'                ELEMENT LISTING SWITCH
         SPACE
IN       DCB   DSORG=PS,MACRF=(GL),DDNAME=IN,EODAD=EOF
         SPACE
OUT      DCB   DSORG=PS,MACRF=(PL),DDNAME=OUT
         SPACE
UPDT     DCB   DSORG=PS,MACRF=(GM),EODAD=EOU,DDNAME=UPDT
         SPACE
MESS     DCB   DSORG=PS,MACRF=(PM),DDNAME=SYSPRINT
         SPACE
LIST     DCB   DSORG=PS,MACRF=(PM),DDNAME=LIST
         SPACE
         END
```

TABLE E

```
         TITLE 'CROSS-REFERENCE VER. 2 - MAIN ROUTINE'
XREF2    CSECT
         SAVE  (14,12)
         BALR  12,0
         USING *,12
         LR    15,13
         L     13,=A(SAVEAREA)
         ST    13,8(15)
         ST    15,4(13)
         USING DUMGP,5
         USING INAR,10
         USING OUTAR,11
         USING SAVEAREA,13
         OPEN  (IN,,OUT,OUTPUT,MESS,OUTPUT)
         SPACE
*           BEGIN  'GET INPUT RECORD' LOOP
         SPACE
GET      MVC   XREFAD,=A(XFLIST)    INITIALIZE X-REF TABLE
         MVC   TP3(3),=XL3'0'       INITIALIZE FLAGS
         MVC   ELAD,=A(ELLIST-4)    INITIALIZE ELEMENT TABLE
         NI    EOS1+1,X'0F'
         OI    EOS2+1,X'F0'
         GET   IN
         ST    1,INAD
         LR    10,1                          ESTABLISH ADDRESSABILITY
         CLI   INSTAN+9,C'7'
         BE    GET
         L     2,NINP
         LA    2,1(2)
         ST    2,NINP
         MVC   HAFWD(2),INDAT
         LH    2,HAFWD
         LTR   2,2
         BNZ   BLOOP
         LA    2,ELLIST
         MVC   0(4,2),=X'40000000'
         ST    2,ELAD
         LA    1,INIDM
         ST    1,FINCH
         MVC   ERNO,=C'19'
         L     15,=A(ERR)
         BALR  14,15
         B     PEND
BLOOP    LA    3,INIDM-1(2)         FIND EXTENT OF IDIOM
         CLI   0(3),C' '
         BNE   FOUCH
         BCT   2,BLOOP              ALL BLANK IF FALL-THRU
         LR    1,3
         ST    3,FINCH              MAKE ONE CHAR BLANK IDIOM
         MVC   ERNO,=C'CB'
         L     15,=A(ERR)
```

```
         BALR  14,15
FINCH    ST    3,FINCH
         LA    2,0
         LA    5,GPI                INITIALIZE STACK WITH REG5
         LA    1,INIDM-1
         SPACE
*          BEGIN 1ST PHASE, NEXT CTL CHAR LOOP
*           1ST PHASE ESTABLISHES LIST OF TEXT AND SYMBOL
*           ELEMENTS
         SPACE
TROOP    LA    3,1(1)               BEGINNING CHAR IN REG3
         L     6,ELAD
         S     6,=A(ELLIST-8)
         C     6,=F'240'
         BNH   STEL
         MVC   ERNO,=C'09'          MORE THAN 60 ELEMENTS
         L     15,=A(ERR)
         BALR  14,15
         B     GET
STEL     STC   6,CUREL
TLEN     L     4,FINCH
         SR    4,3
         SPACE
         EX    4,TRT                LOOK FOR NEXT CTL CHAR, ADDR IN REG1
         SPACE
         BNZ   CC2
         OI    EOS1+1,X'F0'         LAST BYTE(S) IS TEXT
         L     1,FINCH
         LA    1,1(1)
         B     TEXIS
TRT      TRT   0(0,3),TRTABLE
CC2      BNH   CHTEX
         NI    EOS2+1,X'0F'         LAST BYTE IS CONTROL CHAR
CHTEX    CR    1,3
         BNH   NOTEX
TEXIS    CLI   CURGP,0              TEXT BEFORE CONTROL CHAR
         BE    MTEXT
         CLI   CURSUB,0
         BH    MTEXT
         MVC   CURSUB,CUREL         BEGIN SUB-GROUP
         MVI   HSGP,C
MTEXT    ST    3,TEXEL
         LR    4,1
         SR    4,3
         BCTR  4,0
         STC   4,TEXEL
CHTP3    CLI   TP3,C
         BNZ   MTYP3                PRECEDING TYPE 3 CHAR
         CLI   CP,X'95'
         BNE   SEPTEX
         MVC   RADD,TEXEL
```

```
         LA    5,4(5)                  POP UP GROUP DELIMITER, END GROUP
SEPTEX   L     4,ELAD                  TEXT ELEMENT ONLY
         MVC   4(4,4),TEXEL
         LA    4,4(4)
         ST    4,ELAD
         B     ECS1
MTYP3    CLI   TP3,X'95'
         BNE   WTP3
         CLI   TEXEL,0
         BNZ   WTP3
         CLI   0(3),C'R'
         BNE   WTP3
         CLI   GP,0
         BE    MXRR
         MVC   ERNO,=C'02'             REC'D X-REF WITHIN A GROUP
         L     15,=A(ERR)
         BALR  14,15
         B     GET
MXRR     MVI   XREIN,C'F'              ENGLISH 'R' MEANS RECORDED X-REF
         S     5,=F'4'                 PUSH DOWN GROUP DELIMITER
         MVI   GP,X'95'
         MVC   CURGP,CUREL
         MVC   CURSUB,CUREL
         MVI   HSGP,0
WTP3     L     4,ELAD                  TEXT IS X-REF'D
         MVC   4(4,4),=F'0'
         MVC   4(1,4),TP3              MAKE SYMBOL ELEMENT
         MVC   8(4,4),TEXEL            MAKE TEXT ELEMENT
         LA    4,8(4)
         ST    4,ELAD
         CLI   TP3,X'95'
         BE    ZER3
         CLI   TP3,C'|'
         BNE   MXREF
         LA    5,4(5)                  POP UP GROUP DELIMITER, END GROUP
         L     6,RECXR
         LA    6,1(6)
         ST    6,RECXR
         MVC   RADD,TEXEL
         B     ZER3                    NO GEN'D SOURCES FROM XREF2
MXREF    L     6,XREFAD                MAKE X-REF ENTRY
         MVC   0(3,6),HSGP
         MVC   3(1,6),CUREL
         LA    6,4(6)
         ST    6,XREFAD
ZER3     MVI   TP3,0
         B     ECS1
NOTEX    CLI   TP3,0
         BE    ECS1
         MVC   ERNO,=C'03'             TYPE III CHAR NOT FOLLOWED BY TEXT
         L     15,=A(ERR)
```

```
         BALR  14,15
EOS1     NOP   PEND                    END-OF-IDIOM SWITCH
         SPACE
*            DEAL WITH CTL CHAR
         SPACE
         B     *(2)
         B     INVCH
         B     TP1CH
         B     SUBGP
         B     DUAL
         B     LEFTD
         B     RITD
         B     TP3CH
         SPACE
INVCH    MVC   ERNO,=C'01'             INVALID CHARACTER
         L     15,=A(ERR)
         BALR  14,15
         CLI   GP,0
         BE    INVEL
         CLI   CURSUB,0
         BH    INVEL
         MVC   CURSUB,CUREL            BEGIN SUB-GROUP
         MVI   HSGP,0
INVEL    L     4,ELAD                  MAKE SYMBOL ELEMENT
         MVC   4(4,4),=X'40000000'     SUBSTITUTE BLANK
         LA    4,4(4)
         ST    4,ELAD
         B     EOS2
         SPACE
TP1CH    CLI   CURGP,0                 TYPE I CTL CHAR
         BE    TP1EL
         CLI   CURSUB,0
         BH    TP1EL
         MVC   CURSUB,CUREL            BEGIN SUB-GROUP
         MVI   HSGP,0
TP1EL    L     4,ELAD
         MVC   4(4,4),=F'0'
         MVC   4(1,4),0(1)             MAKE CTL CHAR ELEMENT
         LA    4,4(4)
         ST    4,ELAD
         B     EOS2
         SPACE
SUBGP    CLI   CURSUB,0
         BNE   ENSUB
         MVC   ERNO,=C'04'             NO SUB-GROUP TEXT BEFORE DIVIDER
         L     15,=A(ERR)
         BALR  14,15
         B     GET
ENSUB    CLI   0(1),C'/'
         BNE   NONSUB                  MEANS LESS-THAN SIGN
         MVI   CURSUB,0                END SUB-GROUP
```

```
         B       TP2EL
NONSUB   MVC     HSGP,C(1)           INDIVISIBLE SUB-GROUPS
         B       TP2EL
         SPACE
DUAL     CLC     GP,C(1)
         BE      RITD
         SPACE
LEFTD    CLI     CURGP,C
         BE      DOLEF
         CLI     CURSUB,0
         BH      DOLEF
         MVC     CURSUB,CUREL        BEGIN SUB-GROUP
         MVI     HSGP,0
DOLEF    S       5,=F'4'             PUSH DOWN DELIMITER
         MVC     GP,C(1)
         MVC     CURGP,CUREL         BEGIN GROUP
         MVI     CURSUB,0
         B       TP2EL
         SPACE
RITD     CLI     GP,C
         BNE     RUP
         MVC     ERNO,=C'20'         UNMATCHED-RIGHT DELIMITER
         L       15,=A(ERR)
         BALR    14,15
         B       TP2EL
RUP      LA      6,3
FINR     LA      7,T2TAB1(6)
         CLC     C(1,1),0(7)
         BE      RFOUN
         BCT     6,FINR
RFOUN    LA      6,T2TAB2(6)
         CLC     GP,0(6)
         BE      RTOK
         MVC     ERNO,=C'25'         RIGHT DELIMITER DOESN'T MATCH LEFT
         L       15,=A(ERR)
         BALR    14,15
         B       GET
RTOK     LA      5,4(5)              POP UP DELIMITER, END GROUP
         SPACE
TP2EL    L       4,ELAD
         MVC     4(4,4),=F'0'
         MVC     4(1,4),0(1)         MAKE  CTL CHAR ELEMENT
         LA      4,4(4)
         ST      4,ELAD
         B       EOS2
         SPACE
TP3CH    CLI     CURGP,C
         BE      SAV3
         CLI     CURSUB,0
         BH      SAV3
         MVC     CURSUB,CUREL        BEGIN SUB-GROUP
```

```
         MVI   MSGP,0
SAV3     MVC   TP3,0(1)
EOS2     B     TROOP                    END-OF-IDIOM SWITCH
         SPACE 2
*              INITIALIZE SECOND PHASE
         SPACE
PEND     CLI   GP,0
         BNE   FLPEN
         CLI   TP3,0
         BE    CHERR
FLPEN    MVC   ERNO,=C'06'              FLAGS PENDING AT END OF 1ST PHASE
         L     15,=A(ERR)
         BALR  14,15
         B     GET
CHERR    CLI   XREIN,C'R'
         BNE   NORR
         CLI   INSTAN,C'0'
         BE    PH2
         MVC   ERNO,=C'07'              DISCREPANCY WITH STANDARD RECORD
         L     15,=A(ERR)
         BALR  14,15
         B     PH2
NORR     CLI   INSTAN,C'1'
         BE    CHAR
         MVC   ERNO,=C'07'              DISCREPANCY WITH STANDARD RECORD
         L     15,=A(ERR)
         BALR  14,15
CHAR     CLC   XREFAD,=A(XFLIST)
         BE    PH2                      IF EQUAL, NO X-REF'S
         LA    2,INEM+29                GET LENGTH OF LINE ZERO
NONBL    CLI   0(2),C' '
         BNE   SAVL
         BCT   2,NONBL
SAVL     LA    3,INEM
         SR    2,3
         STC   2,LENC
         SPACE
*                2ND PHASE FORMS OUTPUT IDICM USING ENTRIES FROM
*                X-REF TABLE
         SPACE
PH2      LA    2,XFLIST-4
NEWXR    PUT   OUT
         ST    1,OUTAD
         LR    11,1
         MVI   GPE,0
         LA    3,ELLIST-4               REG3,4,5 SCAN ELEMENT LIST WITH BXLE
         LA    4,4
         L     5,ELAD
         LA    6,OUTIDM
         ST    6,PIDM
         LA    2,4(2)                   REG2 POINTS TO X-REF ENTRY
```

```
         C     2,XREFAD
         BL    INXR                X-REF TABLE IS EXHAUSTED WHEN EQUAL
         MVI   OUTTP2,C'0'         '0' MEANS TRANSMIT ORIGINAL IDIOM
         LA    6,STAT5             REG6 IS BRANCH ADDR FOR BXLE
         B     NEXT
INXR     MVI   OUTTP2,C'2'
         LA    6,STAT3A            X-REF OCCURS ALONE
         CLI   1(2),0
         BE    NEXT
         LA    6,STAT1             X-REF OCCURS WITHIN GROUP
         B     NEXT
         SPACE
STAT1    LR    7,3                 STATE 1 MEANS SAVE EVERYTHING WHILE
         S     7,=A(ELLIST-4)      LOOKING FOR START OF GROUP
         EX    7,CGP
         BH    STAT5               IF <=, ELEMENT IS LEFT DELIMITER
         LA    7,4
DEMACH   LA    8,T2TAB2(7)
         CLC   0(1,3),0(8)
         BE    MACHD
         BCT   7,DEMACH
MACHD    IC    7,T2TAB1(7)
         STC   7,GPE               SAVE CORRESPONDING RIGHT DELIMITER
         LA    6,STAT2             SET BRANCH FOR STATE 2
         CLI   0(3),X'95'          ENGLISH AT START OF GROUP MEANS
         BE    NEXT                 REC'D X-REF
         CLI   0(3),C'$'
         BNE   STAT5
         CLI   0(2),0              ZERO MEANS NO INDIVISIBLE SUB-GROUPS
         BE    NEXT                DO NOT PRINT DOLLAR SIGN
         B     STAT5
CGP      CLI   1(2),0              COMPARE GROUP ELEMENT NO.
         SPACE
STAT2    LR    7,3                 STATE 2 MEANS SAVE NOTHING WHILE
         S     7,=A(ELLIST-4)      LOOKING FOR START OF SUB-GROUP
         EX    7,CSG
         BH    NEXT
         LA    6,STAT3
         B     STAT3               CHECK FIRST ELEMENT OF SUB-GROUP
CSG      CLI   2(2),0              COMPARE SUB-GROUP ELEMENT NO.
         SPACE
STAT3    CLI   0(3),C'/'           STATE 3 MEANS SAVE EVERYTHING WHILE
         BNE   CHERD               LOOKING FOR SLASH OR RIGHT DELIMITER
         LA    6,STAT4
         B     NEXT
CHERD    CLC   GPE,0(3)
         BNE   STAT3A              RIGHT DELIMITER IF EQUAL
         LA    6,STAT5
         MVI   GPE,0
         CLI   0(3),C'$'
         BNE   STAT5
```

```
         CLI   C(2),0              ZERO MEANS NO INDIVISIBLE SUB-GROUPS
         BE    NEXT                DO NOT PRINT DOLLAR SIGN
         B     STAT5
         SPACE
STAT3A   LR    7,3                 STATE 3A MEANS SAVE EVERYTHING WHILE
         S     7,=A(ELLIST-4)      CHECKING FOR X-REF
         EX    7,CXR
         BNE   NOXR                X-REF CTL CHAR IF EQUAL
         MVC   TP3,C(3)
         B     NEXT
CXR      CLI   3(2),0              COMPARE X-REF ELEMENT NO.
NOXR     CLI   0(3),64
         BNL   STAT5               TEXT IF LOW
         CLI   TP3,0
         BE    STAT5               X-REF REQUIRED IF NOT =
         MVI   OUTFM,C' '          MAKE OUTPUT LINE ZERO
         MVC   OUTFM+1(29),OUTFM
         LA    7,0
         IC    7,C(3)
         L     8,C(3)
         LA    9,OUTFM
         CLI   TP3,C'*'            DENOTES OBLIQUE CASE X-REF
         BNE   SAVFM
         MVI   OUTFM,C'*'
         LA    9,1(9)
SAVFM    EX    7,MOVIT
         CLI   TP3,C'|'            DENOTES RECORDED X-REF
         BE    MARK
         CLI   TP3,C':'            DENOTES X-REF WITH UPDATED LINE ZERO
         BE    LZXR
         MVI   TP3,0
         B     STAT5
MARK     MVI   OUTTP2,C'1'
LZXR     MVI   TP3,0
         B     NEXT
MOVIT    MVC   0(0,9),0(8)
         SPACE
STAT4    CLC   GPE,C(3)            STATE 4 MEANS SAVE NOTHING WHILE
         BNE   NEXT                LOOKING FOR RIGHT DELIMITER
         LA    6,STAT5
         MVI   GPE,0
         CLI   0(3),C'$'
         BNE   STAT5
         CLI   C(2),0              ZERO MEANS NO INDIVISIBLE SUB-GROUPS
         BE    NEXT                DO NOT SAVE DOLLAR SIGN
         SPACE
STAT5    L     9,PIDM              STATE 5 MEANS SAVE EVERYTHING THRU
         CLI   0(3),64             END OF ELEMENT TABLE
         BL    TEXOUT
         CLI   0(3),C':'
         BNE   SAVCC
```

```
         AR    3,4                    SKIP FOLLOWING TEXT
         B     NEXT                   CANONICAL FORM IS NOT PRINTED
SAVCC    MVC   0(1,9),0(3)            SAVE CTL CHAR
         LA    9,1(9)
         B     MORTX
TEXOUT   LA    7,0
         IC    7,0(3)
         L     8,0(3)
         EX    7,MOVIT                SAVE TEXT
         LA    9,1(7,9)
MORTX    ST    9,PIDM                 NEXT AVAILABLE SPACE IN PIDM
         SPACE
NEXT     BXLE  3,4,0(6)               GO BACK FOR THE NEXT ELEMENT
         SPACE
*            IF FALL-THRU, ELEMENTS OF IDIOM RECONSTITUTED
         SPACE
         CLI   OUTTP2,C'1'
         BE    TOTRX                  '1' MEANS REC'D X-REF
         BL    TOTOR                  '0' (LOW) MEANS ORIGINAL IDIOM
         CLI   XREFIN,C'R'            '2' (HIGH) MEANS GEN'D X-REF
         BL    TOTGX                  RECORDED X-REF PRESENT IF EQUAL
         L     3,PIDM
         MVC   0(5,3),=X'4095099740'  ADD ENGLISH 'R'
         LA    4,0
         IC    4,LENG
         EX    4,MOVC                 ADD LINE ZERO AS A X-REF IN IDIOM
         LA    3,6(4,3)
         ST    3,PIDM
TOTGX    MVI   OUTTP1,C'1'
         L     5,GENXR
         LA    5,1(5)
         ST    5,GENXR
         B     BOXR
MOVC     MVC   5(0,3),INEM
TOTRX    MVI   OUTTP1,C'0'
         L     3,RECXR
         LA    3,1(3)
         ST    3,RECXR
BOXR     MVC   OUTSTAN,INSTAN
         LA    3,0
         IC    3,3(2)
         SRL   3,2
         CVD   3,DEC
         UNPK  OUTEL,DEC
         OI    OUTEL+2,C'0'
         L     15,=A(MAKOUT)
         BALR  14,15
         B     NEWXR                  NEXT X-REF ENTRY
         SPACE
TOTOR    MVI   OUTTP1,C'0'
         CLI   XREFIN,C'R'
```

```
         BNE   SAMSTAN
         MVI   OUTTP1,C'1'
SAMSTAN  MVC   OUTSTAN,INSTAN
         MVC   OUTFM,INFM
         MVC   OUTEL,=C'000'
         L     15,=A(MAKOUT)
         BALR  14,15
         B     GET                    FINISHED WITH INPUT RECORD
         SPACE
*            END-OF-INPUT ROUTINE
         SPACE
EOF      L     2,MAXL                 WRITE STATISTICS MESSAGE
         CVD   2,DEC
         UNPK  STMAX,DEC
         OI    STMAX+2,C'0'
         LA    2,0
         L     3,TOTL
         D     2,NOUT
         CVD   3,DEC
         UNPK  STAVE,DEC
         OI    STAVE+2,C'0'
         L     2,NINP
         CVD   2,DEC
         UNPK  STIN,DEC
         OI    STIN+4,C'0'
         L     2,NOUT
         CVD   2,DEC
         UNPK  STOUT,DEC
         OI    STOUT+4,C'0'
         L     2,GENXR
         CVD   2,DEC
         UNPK  STGEN,DEC
         OI    STGEN+4,C'0'
         L     2,RECXR
         CVD   2,DEC
         UNPK  STREC,DEC
         OI    STREC+4,C'0'
         L     2,ERREC
         CVD   2,DEC
         UNPK  STERC,DEC
         OI    STERC+3,C'0'
         PUT   MESS,STATLN
*            END OF JOB
         CLOSE (IN,,OUT,,MESS)
         STOP
         SPACE
TEXEL    DS    F                      TEMP. TEXT ELEMENT
INAD     DS    F                      INPUT RECORD ADDRESS
OUTAD    DS    F                      OUTPUT RECORD ADDRESS
NINP     DC    F'0'                   TOTAL INPUT RECORDS
RECXR    DC    F'0'                   REC'D X-REF'S
```

```
GENXR    DC    F'0'                    GEN'D X-REF'S                    4
         SPACE
*TRANSLATE AND TEST VALUES IN REG 2:
*              0- TEXT
*              4- INVALID CHARACTER
*              8- TYPE I CTL CHAR, PUNCTUATION, NO SPECIAL FUNCTION
*              12-24 TYPE II CTL CHAR, GROUP DELIMITERS
*              12- SUB-GROUP DIVIDER
*              16- RIGHT OR LEFT DELIMITER
*              20- LEFT DELIMITER
*              24- RIGHT DELIMITER
*              28- TYPE III CTL CHAR, FOLLOWING TEXT MUST BE DEALT WITH
TRTABLE  DC    64X'04'                 INVALID
         DC    X'08'                   BLANK
         DC    10X'04'
         DC    X'00'                   PERIOD
         DC    X'0C'                   LESS-THAN SIGN
         DC    X'14'                   LEFT PARENTHESIS
         DC    X'08'                   PLUS SIGN
         DC    X'10'                   LOGICAL OR
         DC    X'14'                   AMPERSAND
         DC    9X'04'
         DC    X'0C'                   EXCLAMATION POINT
         DC    X'10'                   DOLLAR SIGN
         DC    X'1C'                   ASTERISK
         DC    X'18'                   RIGHT PARENTHESIS
         DC    2X'04'                  SEMI-COLON,LOGICAL NOT,(INVALID)
         DC    X'00'                   HYPHEN
         DC    X'0C'                   SLASH
         DC    9X'04'
         DC    X'C0'                   COMMA
         DC    X'00'                   PERCENT SIGN
         DC    X'1C'                   UNDERSCORE
         DC    X'04'                   GREATER-THAN SIGN,(INVALID)
         DC    X'00'                   QUESTION MARK
         DC    10X'04'
         DC    X'1C'                   COLON
         DC    X'04'                   POUND SIGN,(INVALID)
         DC    X'18'                   AT SIGN
         DC    X'00'                   APOSTROPHE
         DC    X'1C'                   EQUAL SIGN
         DC    X'00'                   QUOTATION MARK
         DC    18X'04'
         DC    X'00'                   X'92' - UPPER CASE RUSSIAN
         DC    2X'04'
         DC    X'1C'                   X'95' - BEGIN ENGLISH
         DC    X'04'
         DC    X'08'                   X'97' - RESUME LOWER CASE RUSSIAN
         DC    41X'04'
         DC    9X'C0'                  A THRU I
         DC    7X'04'
```

```
         DC    7X'00'                  J THRU P
         DC    X'04'                   Q,(INVALID)
         DC    X'00'                   R
         DC    8X'04'
         DC    8X'00'                  S THRU Z
         DC    6X'04'
         DC    2X'04'                  0,1,(INVALID)
         DC    7X'00'                  2 THRU 8
         DC    X'04'                   9 (INVALID)
         DC    6X'04'
         SPACE
XREFAD   DS    F                       POINTER FOR END OF XFLIST
XFLIST   DS    CL40                    TABLE OF X-REF ENTRIES, 4 BYTES EACH
         SPACE
TP3      DS    C                       TYPE III CTL CHAR. FLAG
XREFIN   DS    C                       FLAGS PRESENCE OF REC'D X-REF
LENO     DS    C                       LENGTH OF INPUT LINE ZERO (INFM)
         SPACE
GPE      DS    C                       GROUP DELIMITER FLAG, 2ND PHASE
CUREL    DS    C                       CURRENT ELEMENT NO.
         SPACE
T2TAB1   DC    C')@=$'                 RIGHT DELIMITERS DECODER
         DC    X'95'
T2TAB2   DC    C'(&&=$'                LEFT DELIMITERS DECODER
         DC    X'95'
         SPACE
         DS    8XL4                    GROUP DELIMITER STACK
GP1      DC    XL4'0'
         SPACE
*              "TEMPLATE" FOR GROUP DELIMITER STACK
DUMGP    DSECT
GP       DS    C                       GROUP DELIMITER FLAG
HSGP     DS    C                       INDIVISIBLE SUB-GROUP FLAG
CURGP    DS    C                       GROUP DELIMITER ELEMENT NO.
CURSUB   DS    C                       SUB-GROUP DELIMITER ELEMENT NO.
XREF2    CSECT
         SPACE
STATLN   DC    C'1MAX='
STMAX    DC    C'0000'                 MAXIMUM OUTPUT RECORD LENGTH
         DC    C', AVE='
STAVE    DC    C'000'                  AVERAGE OUTPUT RECORD LENGTH
         DC    C', IN='
STIN     DC    C'00000'                NO. OF INPUT RECORDS
         DC    C', OUT='
STOUT    DC    C'00000'                NO. OF OUTPUT RECORDS
         DC    C', GEN='
STGEN    DC    C'00000'                GEN'D X-REF RECORDS
         DC    C', REC='
STREC    DC    C'00000'                REC'D X-REF RECORDS
         DC    C', ERR='
STERC    DC    C'0000'                 INPUT ERROR RECORDS
```

```
         DC    CL73'.'
         SPACE
         LTORG
         TITLE 'XREF2 - FILL OUTPUT FIELDS COMMON TO ALL ENTRIES'
MAKOUT   CSECT
         USING *,15
         STM   14,5,OUTSV
         DROP  15
         LR    5,15
         USING MAKOUT,5
*             TRANSLATE UPPER CASE RUSSIAN IN LINE-ZERO
         LA    3,OUTFM+29
         LA    1,OUTFM
SUPR     LR    2,3
         SR    2,1
         EX    2,UPTRT
         BZ    ELINB
         MVI   0(1),X'92'
         BH    ELINB
         LA    1,1(1)
         B     SUPR
UPTRT    TRT   0(2,1),UPTAB
*            ELIMINATE INITIAL BLANKS IN IDIOM
ELINB    LA    1,OUTIDM
         L     3,PIDM
         BCTR  3,0                  FINAL POSITION OF IDIOM IN REG3
         CLI   0(1),C' '
         BNE   COVLUP               NO INITIAL BLANK
         LR    2,1
INBL     CLR   2,3
         BNL   AGFIN3               ALL BLANK IF BRANCH
         LA    2,1(2)
         CLI   0(2),C' '
         BE    INBL
         LR    4,3
         SR    4,2
         EX    4,COV1               COVER INITIAL BLANKS
         SR    2,1
         SR    3,2                  NEW FINAL POSITION
*            ELIMINATE X-REF CHARS, TRANSLATE '%', AND REDUCE
*            CONSECUTIVE BLANKS TO ONE
COVLUP   LR    2,3
         SR    2,1
         EX    2,COVTRT             LOOK FOR SPECIAL CHARS
         BZ    CLEN                 NO SPECIAL CHARACTERS
         BH    AGFIN2               LAST CHARACTER
         B     *(2)
         B     COVR                 X'04'
         B     COVBL                X'08'
         B     UPRUS                X'CC'
COVR     LR    2,3
```

```
         SR    2,1
         BCTR  2,0
         EX    2,COV2              DELETE CHARACTER
         BCTR  3,0
         B     COVLUP
COVBL    LA    1,1(1)
         CLI   0(1),C' '
         BNE   COVLUP              IF FALL-THRU, EXTRA BLANKS
         LR    2,1
CBLUP    LA    2,1(2)
         CLR   2,3
         BH    AGFIN1              BRANCH IF IDIOM ENDS IN BLANKS
         CLI   0(2),C' '
         BE    CBLUP
         LR    4,3                 NEXT NON-BLANK WHEN FALL-THRU
         SR    4,2
         EX    4,COV1              COVER EXTRA BLANKS
         SR    2,1
         SR    3,2                 NEW FINAL POSITION
         B     COVLUP
UPRUS    MVI   0(1),X'92'          % -> X'92' FOR UPPER CASE RUSSIAN
         LA    1,1(1)
         B     COVLUP
COVTRT   TRT   0(0,1),COVTAB
COV1     MVC   0(0,1),0(2)
COV2     MVC   0(0,1),1(1)
AGFIN1   BCTR  1,0
AGFIN2   BCTR  1,0
AGFIN3   LR    3,1
*              FILL REMAINING FIELDS
CLEN     LA    3,1(3)
         LR    2,3
         LA    1,OUTIDM
         SR    2,1
         STH   2,HAFWD
         MVC   OUTDAT,HAFWD
         CLI   OUTTP2,C'0'
         BE    LORG
         MVC   0(6,3),=XL6'0'
         LA    3,6(3)
         B     CLOT
LORG     MVC   HAFWD(2),INLN
         LH    1,HAFWD
         AR    1,10
         BCTR  1,0
         MVC   HAFWD(2),INDAT
         LH    2,HAFWD
         LA    2,INIDM(2)
         SR    1,2
CHEL     C     1,=F'256'
         BL    EMV
```

```
         MVC   0(256,3),0(2)
         LA    3,256(3)
         LA    2,256(2)
         S     1,=F'256'
         B     CHEL
EMV      EX    1,MREST
         LA    3,1(1,3)
CLCT     SR    3,11
         C     3,MAXL
         BNH   COMTOT
         ST    3,MAXL                  SAVE MAX LENGTH
COMTOT   L     4,TOTL
         AR    4,3
         ST    4,TOTL                  SAVE TOTAL LENGTH
         STH   3,HAFWD
         MVC   OUTLEN,HAFWD            SAVE RECORD LENGTH
         L     4,NOUT
         LA    4,1(4)
         ST    4,NOUT
         MVC   OUTCFM,OUTFM
         LA    0,30
         LA    1,OUTCFM
         L     15,=A(COMPRESS)
         BALR  14,15                   COMPRESS LINE ZERO (HEADWORD)
         MVC   HAFWD,OUTDAT
         LH    2,HAFWD
         C     2,=F'30'
         BNH   *+8
         LA    2,30
         BCTR  2,0
         MVC   STIM,STIM-1
         EX    2,MOVST
         LA    0,30
         LA    1,STIM
         L     15,=A(COMPRESS)
         BALR  14,15                   FORM SORT FIELD FROM IDIUM
         MVC   OUTSID,STIM
         MVC   OUTID,INID              SAVE ID
         LM    14,5,OUTSV
         RETURN
         SPACE
MREST    MVC   0(0,3),0(2)             SAVE EQUIVALENT AND EXAMPLE
MOVST    MVC   STIM(0),OUTIDM
ALFM     MVC   OUTCFM,0(2)
         SPACE
OUTSV    DC    8F'0'                   OUTPUT ROUTINE SAVE AREA
         SPACE
*              TABLE FOR TRANSLATING '%' TO X'92'
UPTAB    DC    XL109'1'
         DC    XL147'0'
         SPACE
```

```
*            TABLE FOR EDITING IDIOM
*            X'00' - NOT TO BE DELETED
*            X'04' - TO BE DELETED
*            X'08' - BLANK
*            X'0C' - %, INDICATES RUSSIAN CAPITAL LETTER
CONTAB   DC    64X'04'
         DC    X'08'                    BLANK
         DC    10X'04'
         DC    X'0000000C0004000'       .<(+|&
         DC    9X'04'
         DC    X'0000040004040000'      $*);¬-/
         DC    9X'04'
         DC    X'0C0C04040400'          ,%_>?
         DC    10X'04'
         DC    X'04040C000000'          :#@'="
         DC    18X'04'
         DC    X'00'                    X'92' FOR UPPER CASE RUSSIAN
         DC    2X'04'
         DC    X'000400'                X'95,(96),97' FOR ENGLISH
         DC    41X'04'
         DC    9X'0'                    A THRU I
         DC    7X'04'
         DC    7X'0'                    J THRU P
         DC    X'04'                    Q
         DC    X'0'                     R
         DC    8X'04'
         DC    8X'0'                    S THRU Z
         DC    8X'04'
         DC    7X'0'                    2 THRU 6
         DC    7X'04'
         SPACE
         DC    C' '
STAK     DC    CL30' '                  TEMP. IDIOM SORT FIELD
         SPACE
         LTORG
         TITLE 'XREF2 - DELETE SPECIAL CHARACTERS FROM RUSSIAN'
COMPRESS CSECT
         USING *,15
         STM   1,4,CMPSV
         DROP  15
         LR    4,15
         USING COMPRESS,4
         LR    2,0
         BCTR  1,0
C9BLUP   LA    3,0(1,2)                 FIND FINAL NON-BLANK CHAR
         CLI   0(3),C' '
         BNE   C9NONB
         BCT   2,C9BLUP
         B     C9OUT                    ALL BLANK
C9NONB   LA    1,1(1)
         LNR   2,1
```

```
         AR    2,3
C9BACK   EX    2,C9TRT                 FIND SPECIAL CHARACTER
         BZ    C9OUT                   FINISHED
         CR    1,3
         BE    C9POUT                  LAST CHAR IS SPECIAL
         LNR   2,1
         AR    2,3
         BCTR  2,0
         EX    2,C9CUVR                SHIFT OVER SPECIAL CHAR
         MVI   0(3),C' '
         BCTR  3,0
         B     C9BACK                  NEXT SPECIAL CHAR
C9POUT   MVI   0(3),C' '
C9OUT    LM    1,4,CMPSV
         RETURN
         SPACE
C9CUVR   MVC   0(0,1),1(1)
C9TRT    TRT   0(0,1),C9SPEC
         SPACE
CMPSV    DC    4F'0'                   REG SAVE AREA
         SPACE
C9SPEC   DC    193X'01'                ZERO ENTRIES FOR KEYPUNCH CODE
         DC    XL16'0101010101010101'
         DC    XL17'01010101010101010'
         DC    XL16'0101010101010101'
         DC    XL14'01010101010101'
         SPACE
         LTORG
         TITLE 'AREF2 - WRITE ERROR MESSAGE'
ERR      CSECT
         USING *,15
         STM   14,7,ESV
         DROP  15
         LR    7,15
         USING ERR,7
         LR    3,1                     GET POSITION IN IOIOM WHEN ERROR
         LA    2,INIOM-1               TURNS UP
         SR    3,2
         CVD   3,DEC
         UNPK  ERPOS,DEC
         OI    ERPOS+2,C'0'
         MVC   ERID,INID
         MVC   ERSTAN,INSTAN
         MVC   ERFM,INFM
         AP    NOMES,=P'1'
         CP    NOMES,=P'8'
         BNH   SAMP
         PUT   MESS,HDLINE             START NEW PAGE
         ZAP   NOMES,=P'1'
SAMP     PUT   MESS,ERLINE
         L     2,ERREC
```

```
         LA    2,1(2)
         ST    2,ERREC
*             WRITE IDIOM IN HEXADECIMAL
         CI    MESW+1,X'FC'
         LA    2,INIDM
         MVC   HXLINO(2),=C'0 '
MESLN    LA    3,1
         MVC   HXLINO+2(256),HXLINO+1      BLANK PRINT LINES
         MVC   HXLINO+258(141),HXLINO+257
LTEST1   C     2,FINCH
         BNH   LTEST2                      FINISHED IF HIGH
         NI    MESW+1,X'0F'
         B     RIMES
LTEST2   C     3,=F'100'
         BH    RIMES                       PRINT LINES FILLED IF HIGH
         LA    4,HXLINO+5(3)
         MVC   0(1,4),0(2)                 SAVE CHARACTER IN FIRST LINE
         LA    5,0
         IC    5,0(2)
         LR    6,5
         N     6,=F'15'                    2ND HEX DIGIT IN REG6
         SRL   5,4                         1ST HEX DIGIT IN REG5
         LA    4,HXLIN1+5(3)
         IC    5,HEXTAB(5)
         STC   5,0(4)                      STORE 1ST HEX CHARACTER
         LA    4,HXLIN2+5(3)
         IC    6,HEXTAB(6)
         STC   6,0(4)                      STORE 2ND HEX CHARACTER
         LA    2,1(2)
         LA    3,1(3)
         B     LTEST1
RIMES    PUT   MESS,HXLINO                 WRITE IDIOM AND HEX IMAGES
         PUT   MESS,HXLIN1
         PUT   MESS,HXLIN2
MESW     B     MESLN                       END-OF-OPERATION SWITCH
         LM    14,7,ESV
         RETURN
         SPACE
ESV      DC    10F'0'                      ERROR ROUTINE SAVE AREA
         SPACE
HLLINE   DC    CL133'1ERROR  POSITION    ID     STAN. RECORD   LINE ZERO
               0'
HXLINO   DS    CL133                       IDIOM IMAGE
HXLIN1   DS    CL133                       FIRST HEX DIGITS
HXLIN2   DS    CL133                       2ND HEX DIGITS
         SPACE
HEXTAB   DC    C'0123456789ABCDEF'         HEX TRANSLATE TABLE
NOMES    DC    P'3'                        LINE COUNT
         SPACE
         LTORG
         TITLE 'XREF2 - GENERAL CONSTANTS AND AREAS'
```

```
SAVEAREA CSECT
         DC    18F'0'                    SAVE AREA
DEC      DS    D                         DECIMAL CONVERSION
HALFWD   DC    F'0'                      PROVIDES A HALF-WORD BOUNDARY + ZERO
NOUT     DC    F'0'                      TOTAL OUTPUT RECORDS
MAXL     DC    F'0'                      MAX. RECORD LENTH
TOTL     DC    F'0'                      TOTAL LENGTH OF ALL OUTPUT RECORDS
ERREC    DC    F'0'                      ERROR RECORDS
FINCH    DS    F                         FINAL NON-BLANK CHAR IN INPUT IDIOM
PIDM     DS    F                         END OF OUTPUT IDIOM + ONE
RADD     DS    F                         ADDR OF X-REFERENCE
         SPACE
MESS     DCB   DSORG=PS,MACRF=(PM),RECFM=FA,BLKSIZE=133,DDNAME=SYSPRINT
         SPACE
OUT      DCB   DSORG=PS,MACRF=(PL),DDNAME=OUT
         SPACE
IN       DCB   DSORG=PS,MACRF=(GL),DDNAME=IN,EODAD=EOF
         SPACE
ELAD     DS    F                         POINTER TO END OF ELLIST
ELLIST   DS    66F                       STRING OF TEXT OR SYMBOL ELEMENTS
         SPACE
ERLINE   DC    C'-'                      ERROR MESSAGE
         DC    C' '
ERNO     DC    C'00'
         DC    CL6' '
ERPOS    DC    C'000'
         DC    CL4' '
ERID     DC    C'000000000'
         DC    CL3' '
ERSTAN   DC    C'000000000'
         DC    CL4' '
ERFM     DC    CL30' '
         DC    CL62' '
*              INPUT RECORD AREA
INAR     DSECT
INLN     DS    CL4                       VARIABLE RECORD LENGTH
INID     DS    CL9                       ID
INSTAN   DS    CL12                      STANDARD RECORD
INFM     DS    CL30                      LINE ZERO
INDAT    DS    CL2                       BEGINNING OF DATA
INIDM    DS    CL1                       BEGINNING OF IDIOM
*              OUTPUT RECORD AREA
OUTAR    DSECT
OUTLN    DS    CL4                       VARIABLE RECORD LENGTH
OUTTP1   DS    CL1                       ENTRY TYPE, 'R' PRESENT OR ABSENT
OUTCFM   DS    CL30                      COMPRESSED LINE ZERO, FOR SORT
OUTSID   DS    CL15                      ID SORT ON BEGINNING OF IDIOM
OUTTP2   DS    CL1                       ENTRY TYPE, ORIGINAL OR GENERATED
OUTID    DS    CL9                       ID
OUTEL    DS    CL3                       X-REF ELEMENT NO.
OUTSTAN  DS    CL12                      STANDARD RECORD
```

```
OUTFM    DS    CL3C          LINE ZERO
OUTDAT   DS    CL2           BEGINNING OF DATA
OUTIDM   DS    CL1           BEGINNING OF IDIOM
         END
```

TABLE F

```
         TITLE 'IDIOM FORMATTER - MAIN ROUTINE'
FORMID   CSECT
*            PRINT CONSTANTS
SEGL     EQU   62                      COLUMN WIDTH
IDT      EQU   3                       INDENTATION
NOCL     EQU   2                       NO. OF COLUMNS
LINL     EQU   NOCL*SEGL               LENGTH OF ARRAY ROW
NOLN     EQU   94                      MAX. LINES PER PAGE
SPC      EQU   0                       1 - SPACING; 0 - NO SPACING
         SPACE
         SAVE  (14,12)
         BALR  12,0
         USING *,12
         USING INAR,10
         USING SAVE,13
         LR    15,13
         L     13,SAVAD
         ST    13,8(15)
         ST    15,4(13)
         OPEN  (IN,,MESS,OUTPUT,OUT,OUTPUT)
*            GET A NEW ENTRY
GET      GET   IN
         LR    10,1
         AP    PIN,=P'1'               INCREMENT ENTRY COUNTER
         CLI   INTP1,C'0'              ZERO MEANS SOURCE ENTRY
         BNE   CRF                     ONE MEANS CROSS-REFERENCE
         AP    POS,=P'1'
         B     STEC
CRF      CLI   INTP2,C'0'              ZERO MEANS ORIGINAL ENTRY
         BNE   GRF                     TWO MEANS GENERATED
         AP    POR,=P'1'
         B     STEC
GRF      AP    PGR,=P'1'
*            SET UP POINTERS FOR EACH SECTION IN ENTRY
STEC     LA    2,SECTAB
         LA    3,INDAT
SCLUP    CLC   =F'0',0(2)
         BE    MODID
         MVC   HALFWD,0(3)
         LA    3,2(3)
         MVC   4(4,2),FULWD
         ST    3,0(2)
         A     3,FULWD
         LA    2,8(2)
         B     SCLUP
*            CAPITALIZE FIRST LETTER OF IDIOM
MODID    MVC   TIDSAV,SECTAB
         LA    1,SECTAB
         L     15,=A(EDIM)
         BALR  14,15
         LA    9,SECTAB                ADDR OF SECTION PTR IN REG9
```

```
*                GET POINTERS FOR NEXT SECTION
GETSEC   CLC    =F'0',0(9)
         BE     GET                     END OF TABLE
         CLC    =F'0',4(9)
         BE     ENDSEC                  EMPTY SECTION
         LA     2,SECTAB
         LR     3,9
         SR     3,2
         SRL    3,1
         ST     3,SECNO                 PARAMETER FOR "STORAD"
         LA     4,BID(3)
         L      5,0(4)
         LA     5,1(5)                  INCREMENT SECTION COUNTER
         ST     5,0(4)
         MVC    SEGAD(8),0(9)
         MVC    INDENT,LIDT             1ST LINE INDENTATION
         MI     SWEG+1,X'0F'
*             TAKE A SEGMENT OF A SECTION
GETSEG   LA     6,SEGL-IDT-1
         S      6,INDENT
         C      6,LREM
         BL     MORSEC
*            SECTION IS FINISHED HERE
         OI     SWEG+1,X'F0'
         L      2,LREM
         BCTR   2,0
         B      LASEG
MORSEC   L      0,SEGAD
         LR     1,6
         L      15,=A(DIVID)            FIND THE DIVIDING POINT IN SECTION
         BALR   14,15
         LR     2,1
LASEG    L      1,SECNO
         L      15,=A(STORAD)           GET THE POSITION IN THE PRINT PAGE
         BALR   14,15
         A      1,INDENT
         L      3,SEGAD
         EX     2,SEGMOV                MOVE THE SEGMENT TO THE PRINT PAGE
         MVC    SECNO,=F'16'
         MVC    INDENT,=F'0'
SWEG     NOP    ENDSEC                  END-OF-SECTION SWITCH
         LA     2,1(2)                  FIND THE NEXT ADDRESS IN THE SECTION
         L      3,LREM
         SR     3,2
         ST     3,LREM
         A      2,SEGAD
         ST     2,SEGAD
         B      GETSEG
ENDSEC   LA     9,8(9)                  NEXT SECTION POINTER
         B      GETSEC
*            END OF INPUT
```

```
EOF     L       15,=A(PUTL)             WRITE LAST PAGE
        BALR    14,15
*           WRITE STATISTICS MESSAGE
        UNPK    ZIN,PIN
        OI      ZIN+4,C'0'
        UNPK    ZOUT,POUT
        OI      ZOUT+3,C'0'
        UNPK    ZOS,POS
        OI      ZOS+4,C'0'
        UNPK    ZGR,PGR
        OI      ZGR+4,C'0'
        UNPK    ZGR,PGR
        OI      ZGR+4,C'0'
        L       2,BID
        CVD     2,DEC
        UNPK    DID,DEC
        OI      DID+4,C'0'
        L       2,BEN
        CVD     2,DEC
        UNPK    DEN,DEC
        OI      DEN+4,C'0'
        L       2,BEX
        CVD     2,DEC
        UNPK    DEX,DEC
        OI      DEX+4,C'0'
        L       2,B4
        LTR     2,2
        BZ      WRAT
        CVD     2,DEC
        UNPK    D4,DEC
        OI      D4+4,C'0'
WRAT    PUT     MESS,STAT1
        PUT     MESS,STAT2
        CLOSE   (IN,,MESS,,OUT)
        STOP
        SPACE
SEGMOV  MVC     0(0,1),0(3)             PUTS SEGMENT IN PAGE
        SPACE
SAVAD   DC      A(SAVE)                 ADDR OF SAVE AREA AND CONSTANTS
TIDSAV  DS      CL8                     ORIGINAL IDIOM POINTER
INDENT  DS      F                       INDENTATION INDICATOR
*           TABLE OF SECTION POINTERS - 1ST WD-ADDR, 2ND WD-LENGTH.
SECTAB  DC      8F'-1'                  INITIAL NON-ZERO VALUES
        DC      F'0'                    ZERO MARKS END OF TABLE
*           BOUNDARIES
FULWD   DS      0F
        DC      H'0'
HAFWD   DC      H'0'
*           SEGMENT POINTER
SEGAD   DS      F                       ADDR OF CURRENT SEGMENT
LREM    DS      F                       LENGTH OF REST OF SECTION
```

```
*                COUNTERS
BID       DC     F'0'                    ENTRIES WITH IDIOM
BEN       DC     F'0'                    ENTRIES WITH ENGLISH EQUIVALENT
BEX       DC     F'0'                    ENTRIES WITH LANGUAGE EXAMPLE
B4        DC     F'0'                    UNUSED 4TH SECTION
PIN       DC     PL3'0'                  INPUT ENTRIES
POS       DC     PL3'0'                  COUNT ORIGINAL SOURCES
POR       DC     PL3'0'                  COUNT ORIGINAL CROSS-REFERENCES
PGR       DC     PL3'0'                  COUNT GENERATED CROSS-REFERENCES
*             TITLE LINE FOR STATISTICS MESSAGE
STAT1     DC     CL133'1ENTRIES PAGES  ORG SRC  ORG REF  GEN REF  IDIOMS E
                EQUIV  EXMPL'
*             STATISTICS TOTALS
STAT2     DC     CL2' '
ZIN       DC     C'00000'
          DC     CL2' '
ZOUT      DC     C'0000'
          DC     CL4' '
ZOS       DC     C'00000'
          DC     CL4' '
ZOR       DC     C'00000'
          DC     CL4' '
ZGR       DC     C'00000'
          DC     CL3' '
DID       DC     C'00000'
          DC     CL3' '
DEN       DC     C'00000'
          DC     CL2' '
DEX       DC     C'00000'
          DC     CL2' '
D4        DC     CL5' '
          DC     CL63' '
          TITLE 'IDIOM FORMATTER - GET OUTPUT STORE ADDRESS'
STORAD    CSECT
          USING *,15
          STM    14,6,STAVE
          DROP   15
          LR     6,15
          USING STORAD,6
FIRSW     B      INITL                   INITIALIZATION SWITCH
          L      2,CURLN                 CURRENT LINE NO. ( X LINL) IN REG2
          L      3,COLNO                 ADDR OF COLUMN IN REG3
          LA     2,LINL(2)
          C      2,=A((NOLN-17)*LINL)
          BNH    BRTAB
*             END OF COLUMN HERE
          C      3,=A(LINES+((NOCL-1)*SEGL))
          BL     NEXOL
*             PAGE IS FILLED HERE
          L      15,=A(PUTL)             WRITE PAGE
          BALR   14,15
```

```
*                BLANK PAGE
ENTRI    L       3,=A(LINES)
         MVC     0(LINL,3),BLANK
         LA      4,LINL
         L       5,=A(LINES+((NOLN-2)*LINL))
BLUP     MVC     LINL(LINL,3),0(3)
         BXLE    3,4,BLUP
         MVC     LASLN,=F'0'             INITIALIZE LAST LINE COUNTER
         L       3,=A(LINES)             AT 1ST COLUMN
         B       NEWCOL
NEXOL    LA      3,SEGL(3)               NEXT COLUMN
NEWCOL   ST      3,COLNO
         LA      2,0                     AT 1ST LINE
BRTAB    L       1,SECNO
         B       *+4(1)
         B       SEC1
         B       SEC2
         B       SEC3
         B       SEC4
         B       SAMSEC                  CONTINUATION IN SAME SECTION
SEC2     MVI     URP,X'95'               SEC. 2 IS ENGLISH EQUIVALENT
         B       SAMSEC
SEC4     MVI     URP,X'95'               SEC. 4 IS NOT USED AT PRESENT
         B       SAMSEC
SEC3     MVI     URP,X'97'               SEC. 3 IS LANGUAGE EXAMPLES
SAMSEC   LTR     2,2
         BZ      LIN1A
         LA      2,SPC*LINL(2)
         B       PREBL
*             AT 1ST LINE HERE
LIN1A    C       3,=A(LINES)
         BNE     PREBL
*             AT 1ST COL, 1ST LINE HERE; WRITE HEADWORD
         LA      4,0(2,3)
         MVC     0(31,4),SAVEM-1         MOVE HEADWORD AND URP CTL CHAR
         LA      5,30(4)
NOBLI    CLI     0(5),C' '               FIND END OF HEADWORD
         BNE     *+8
         BCT     5,NOBLI
         MVC     1(8,5),=X'85C30D6D5E37DC44B' ADD "CONT'D."
         LA      2,SPC*LINL(2)           SKIP A SPACE
         B       NEXLN                   NEXT LINE
SEC1     MVI     URP,X'97'               SEC1 (IDIOM) BEGINS ENTRY
         CLC     SAVEM,INEM
         BE      SAME
*             NEW HEADWORD HERE
         LTR     2,2
         BZ      NEWD
         LA      2,2*SPC*LINL(2)         SPACE 2 IF NOT AT 1ST LINE
NEWD     MVC     SAVEM,INEM
         LA      4,0(2,3)
```

```
         MVC   0(31,4),SAVEM-1
         LA    2,SPC*LINL(2)
         LA    2,LINL(2)
         B     STANL
*             SAME HEADWORD
SAHD     LTR   2,2
         BZ    ATL1                    ZERO MEANS 1ST LINE
         LA    2,2*SPC*LINL(2)
         B     STANL
ATL1     C     3,=A(LINES)
         BNE   STANL
*            AT 1ST COL, 1ST LINE HERE; WRITE HEADWORD
         LA    4,0(2,3)
         MVC   0(31,4),SAVEM-1
         LA    5,30(4)
NOBL2    CLI   0(5),C' '
         BNE   *+8
         BCT   5,NOBL2
         MVC   1(8,5),=X'B5C306D5E37DC44B' ADD "CONT'D."
         LA    2,SPC*LINL(2)           SKIP A SPACE
         LA    2,LINL(2)
*            WRITE STANDARD RECORD
STANL    LA    4,0(2,3)
         MVI   0(4),X'95'
         A     4,LIDT
         LA    4,1(4)
         MVC   IDT(1,4),INSTAN1
         MVC   IDT+2(1,4),INSTAN2
         MVC   IDT+4(2,4),INSTAN3
         MVC   IDT+7(3,4),INSTAN4
         MVC   IDT+11(1,4),INSTAN5
         MVC   IDT+13(1,4),INSTAN6+1
         MVC   IDT+17(9,4),INID        WRITE ID
         MVC   IDT+28(3,4),INEL
         LA    2,SPC*LINL(2)
NEXLN    LA    2,LINL(2)               NEXT LINE
PREBL    LA    4,0(2,3)
         MVC   0(1,4),URP              URP CTL CHAR FOR CURRENT SECTION
         LA    4,1(4)
SALN     ST    2,CURLN
         C     2,LASLN
         BNH   RETAD
         ST    2,LASLN                 SAVE LAST LINE
RETAD    LA    1,IDT(4)                RETURN SEGMENT STORE ADDR IN REG1
         LM    14,0,STAVE
         LM    2,0,STAVE+16
         RETURN
*            INITIALIZE THINGS
INITL    NI    FIRSW+1,X'0F'
         B     ENTRI
         SPACE
```

```
STAVE    DC     9F'0'                  SAVE AREA
CURLN    DS     F                      CURRENT LINE NO. ( X LINL)
CCLNO    DS     F                       CURRENT COLUMN ADDR
*             BLANK LINE TO BLANK THE PAGE
BLANK    DS     CCL(LINL)
         DC     CL(LINL)' '
*             HEADWORD COMPARISON
         DC     X'96'                  UPPER CASE RUSSIAN
SAVEM    DC     CL30' '
         SPACE
URP      DC     X'97'                  UPP CHAR FOR CURRENT SECTION
         TITLE 'IDIOM FORMATTER - CAPITALIZE FIRST WORD IN IDIOM'
EDIM     CSECT
         USING *,15
         STM    14,6,EDSAV
         DROP   15
         LR     6,15
         USING EDIM,6
*             ADDR OF CURRENT CHAR IN REG3, REMAINING LENGTH IN REG4
         LM     3,4,C(1)
         BCTR   4,0
         LA     5,0(3,4)
         ST     5,ENCH
         L      5,=A(STOPG)            CURRENT SAVE ADDR IN REG5
*             SCAN IDICM FOR FIRST TEXT CHARACTER BYTE-BY-BYTE
         LA     2,0
         IRT    0(1,3),TEXTAB
         BZ     NECH
         MVI    0(5),X'92'
         LA     5,1(5)
         B      GREM
NECH     MVC    0(1,5),0(3)
         LA     5,1(5)
         C      3,ENCH
         BNL    RTN
         IRT    1(1,3),TEXTAB
         BZ     CNT                    TEXT CHAR IF NON-ZERO
*             CHECK FOR PRECEDING URP CHAR - ENGLISH OR CAPITAL
         CLI    0(3),X'92'
         BE     DUN                    ALREADY CAPITALIZED
*             IRT VALUE IN REG2 IS F'4' OR F'8'
         B      *(2)
         B      CAP                    4 MEANS CHAR NOT NUMERIC
         CLI    0(3),X'95'             8 MEANS NUMERIC
         BE     CNT                    ENGLISH NUMERIC IS GRAPHIC NUMERAL
*             INSERT URP CAPITALIZATION
CAP      MVI    0(5),X'92'
         LA     5,1(5)
DUN      LA     3,1(3)
GREM     L      4,ENCH
         SR     4,3
```

```
         B       STOLP                   SAVE REST OF IDIOM
*              NEXT CHARACTER
CNT      LA      3,1(3)
         B       NECH
*              SAVE REST OF IDIOM BLOCK-BY-BLOCK
STOLP    C       4,=F'256'
         BL      STEND
         MVC     0(256,5),0(3)
         LA      5,256(5)
         LA      3,256(3)
         S       4,=F'256'
         B       STOLP
STEND    EX      4,LASMV                 SAVE FINAL SEGMENT OF IDIOM
         LA      5,1(4,5)
RTN      L       4,=A(STORG)
         SR      5,4
         L       1,EDSAV+12
         STM     4,5,0(1)                RETURN NEW PARAMETERS TO "SECTAB"
         LM      14,6,EDSAV
         RETURN
         SPACE
LASMV    MVC     0(0,5),0(3)
         SPACE
EDSAV    DC      9F'0'                   REG SAVE AREA
         SPACE
*              TRANSLATE AND TEST TABLE FOR TEXT CHARACTERS
TEXTAB   DC      193X'00'
         DC      9X'04'                  A THRU I
         DC      7X'00'
         DC      7X'04'                  J THRU P
         DC      X'00'
         DC      X'04'                   R
         DC      8X'00'
         DC      8X'04'                  S THRU Z
         DC      8X'00'
         DC      7X'08'                  2 THRU 8
         DC      7X'00'
         SPACE
ENCH     DS      F                       END OF IDIOM ADDR
         SPACE
STORG    CSECT
         DS      CL1024                  IDIOM SAVE AREA
         TITLE 'IDIOM FORMATTER - SEGMENT DIVIDE ROUTINE'
DVID     CSECT
         USING *,15
         STM     14,6,DSAV
         DROP    15
         LR      6,15
         USING DVID,6
*              SEGMENT BEGIN ADDR IN REG3, LENGTH IN REG4
         LM      3,4,DSAV+8
```

```
         LA    2,0
         BCTR  4,0
BLIP     LA    5,0(3,4)
         TRT   0(1,5),SCTAB
         B     *(2)
         B     BAKUP                   TABLE VALUE X'04'
         B     RIGHT                   X'08'
         B     DIVOK                   X'0C'
*            NON-DIVIDE CHARACTER
BAKUP    BCT   4,BLIP
*            NO DIVIDE CHAR FOUND HERE
         L     4,DSAV+12               FULL SEG. LEN.
         BCTR  4,0
         B     DIVOK
*            RIGHT DELIMITER
RIGHT    CLI   1(5),C'('
         BE    BAKUP                   NESTED GROUP
         CLI   1(5),C'='
         BE    BAKUP     NESTED GROUP
*            DIVIDE AT THIS CHARACTER
DIVOK    ST    4,DSAV+12               RETURN EX LENGTH IN REG1
         LM    14,6,DSAV
         RETURN
         SPACE
DSAV     DC    9F'0'                   REG SAVE AREA
         SPACE
*            TRANSLATE AND TEST TABLE FOR DIVIDE CHARACTER
SCTAB    DC    64X'04'
         DC    X'0C'                   BLANK
         DC    26X'04'
         DC    X'08'                   RIGHT PARENTHESIS
         DC    3X'04'
         DC    X'0C'                   SLASH
         DC    26X'04'
         DC    X'08'                   RIGHT BRACKET (])
         DC    131X'04'
         TITLE 'IDIOM FORMATTER - WRITE PAGE ROUTINE'
PUTL     CSECT
         USING *,15
         STM   14,6,PSAV
         DROP  15
         LR    6,15
         USING PUTL,6
         AP    PLOT,=P'1'              INCREMENT PAGE COUNTER
         MVI   CCH,C'1'
         SR    5,5
*            NEXT-LINE LOOP
LLOOP    MVC   LINE,LINE-1
         MVC   COLPT,=A(COLTAB)
         MVC   SHFT,=F'0'
         ST    5,LINP
```

```
         A     5,=A(LINES)          GET ADDR OF SEGMENT
*           NEXT-COLUMN LOOP
COLUP    L     1,COLPT
         L     1,0(1)
         AL    1,SHFT
         LA    2,SEGL-1(1)
         C     2,=A(ENLN)
         BNH   MOK
         L     3,=A(ENLN)
         LR    2,3
         SR    3,1
         BM    *+8                  INITIAL ADDR TOO HIGH
         EX    3,MSEG
         MVC   UID,INID
         MVC   UFM,INFM
         PUT   MESS,MORP
         MVI   LINE,C' '
         PUT   MESS,LINE
         MVI   MORP,C' '
         B     RITE
MOK      MVC   0(SEGL,1),0(5)
         ST    2,SEND
         BCTR  1,0
         SR    4,4
*           SCAN FOR ORP-DELETE CHARACTERS
SCOP     LA    3,1(1)
         L     2,SEND
         SR    2,3
         EX    2,SCORP
         BZ    SHAD
         LA    4,1(4)
         BNH   SCOP                 COND CODE 2 MEANS LAST CHAR.
SHAD     A     4,SHFT
         ST    4,SHFT
         LA    5,SEGL(5)
         L     1,COLPT
         TM    0(1),X'80'
         LA    1,4(1)
         ST    1,COLPT
         BZ    COLUP                NON-ZERO MEANS LAST COLUMN
         MVC   LINE(1),CCH
RITE     PUT   OUT,LINE
         MVI   CCH,C' '
         L     5,LINP
         LA    5,LINE(5)
         C     5,LASEN
         BNH   LLOP
         MVC   LINE,LINE-1
         MVI   LINE,C'0'
         PUT   OUT,LINE
         MVC   LINE+59(5),=5X'20'
```

```
         ED    LINE+56(0),PUOT
         PUT   OUT,LINE
         LM    14,6,PSAV
         RETURN
         SPACE
PSAV     DC    9F'0'
         SPACE
SCURP    TRT   0(0,3),TURP          SCAN INSTRUCTION
MSEG     MVC   0(0,1),0(5)
         SPACE
SEND     DS    F                    FINAL SCAN ADDRESS
SHFT     DS    F                    CUMULATIVE LINE SHIFT
LINP     DS    F                    NO. OF CURRENT LINE
*              TABLE OF URP DELETE CHARACTERS
TURP     DC    XL147'1'             X'92'
         DC    XL2'0'
         DC    X'010101'            X'95',X'96',X'97'
         DC    XL104'0'
         SPACE
CCH      DC    C'1'                 CARRIAGE CONTROL
*           COLUMN POINTERS
COLPT    DS    F                    POINTS TO CURRENT COLUMN ADDR
COLTAB   DC    A(LINE+1)            ADDR OF 1ST COLUMN
         DC    AL1(128)
         DC    AL3(LINE+1+SEGL+1)   ADDR OF 2ND COLUMN
         SPACE
         DC    C' '
LINE     DC    CL143' '             PRINT LINE PLUS SHIFT SPACE
ENLN     EQU   *-1
         SPACE
MURP     DC    C'1'
         DC    C'23 - '
UID      DC    CL9' '
         DC    C' '
UFM      DC    CL30' '
         DC    CL87' '
         TITLE 'IDIOM FORMATTER - COMMON CONTROL SECTION'
SAVE     CSECT
         DC    18F'0'               SAVE AREA
DEC      DS    D                    DECIMAL CONVERSION BOUNDARY
SECND    DS    F                    SECTION BRANCH VALUE
LASLN    DS    F                    LAST LINE NO. ON PAGE ( X LINE)
LIDT     DC    F'0'                 1ST LINE INDENTATION CONSTANT
         SPACE
INAR     DSECT
INLN     DS    CL4                  OS/360 RECORD LENGTH
INTP1    DS    CL1                  REF. OR SOURCE
INCFM    DS    CL30                 COMPRESSED HEADWORD
INSID    DS    CL15                 FIRST CHAR'S OF IDIOM
INTP2    DS    CL1                  GEN'D OR NOT
INID     DS    CL9                  ID
```

```
INEL     DS    CL3                    ELEMENT NO.
INSTAN   DS    CL12                   STANDARD RECORD
INSTAN1  DS    C
INSTAN2  DS    C
INSTAN3  DS    CL2
INSTAN4  DS    CL3
INSTAN5  DS    C
INSTAN6  DS    CL2
INSTAN7  DS    CL2
INFM     DS    CL30                   HEADWORD
INDAT    DS    CL1                    BEGINNING OF IDIOM, EQUIV., ETC.
SAVE     CSECT
         SPACE
IN       DCB   DSORG=PS,MACRF=(GL),EODAD=ECF,DDNAME=IN
         SPACE
OUT      DCB   DSORG=PS,MACRF=(PM),DDNAME=OUT
         SPACE
MESS     DCB   DSORG=PS,MACRF=(PM),RECFM=FA,BLKSIZE=133,DDNAME=SYSPRINT
         SPACE
PCOUT    DC    PL3'0'                 PAGE COUNTER
         SPACE
         LTORG
         SPACE
*              ARRAY OF LINES AND COLUMNS
LINES    CSECT
         DS    (NCEN)CL(LINE)
         END
```